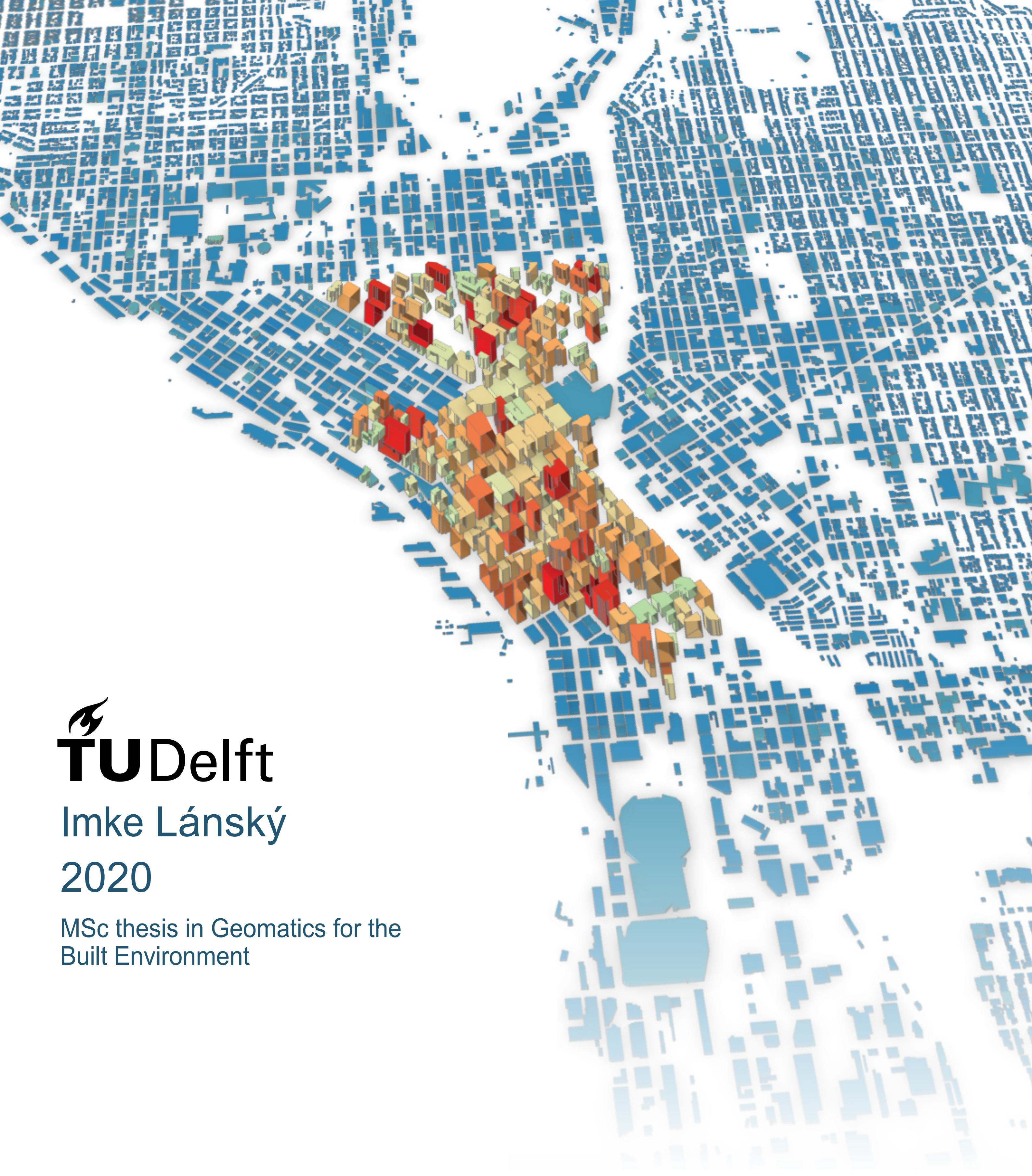


TU Delft

Imke Lánský

2020

MSc thesis in Geomatics for the Built Environment

Height Inference for all USA Building Footprints in the Absence of Height Data

MSc thesis in Geomatics

Height Inference for all USA Building Footprints in the Absence of Height Data

Imke Lánský

June 2020

A thesis submitted to the Delft University of Technology in partial fulfilment of the requirements for the degree of Master of Science in Geomatics

Imke Lánský: *Height Inference for all USA Building Footprints in the Absence of Height Data* (2020)

The work in this thesis was carried out in the:

3D geoinformation group
Department of Urbanism
Faculty of the Built Environment & Architecture
Delft University of Technology

Supervisors: Dr. H. Ledoux
B. Dukai
Dr. F. Biljecki
Co-reader: Prof. dr. J.E. Stoter

Abstract

In recent years, the demand for 3D spatial information and 3D city models has increased, as they support and allow many different applications, e.g. noise simulations, energy demand estimations, and shadow analysis. Constructing a city model with 3D buildings requires elevation data (such as LiDAR or Digital Terrain Models), but unfortunately, data of sufficient quality is often unavailable. This thesis focuses on the use of machine learning methods to estimate the height of building footprints and thus bypassing the use of elevation data completely. Three different methods are tested and compared: Random Forest Regression (RFR), Multiple Linear Regression (MLR), and Support Vector Regression (SVR).

A case study is performed for the conterminous United States of America (USA) because of its availability of a nation-wide building dataset, containing roughly 125 million building footprints. The high diversity in urban layouts is considered, where a distinction is made between Central Business Districts (CBDs) in cities and all other regions (e.g. suburbs and rural areas). All building footprints are characterised by nine features derived from their geometry, which are then used (in several combinations) in the model training and predicting stages. Furthermore, the influence of additional features — including census and cadastral data — on the results of the building height predictions is analysed for the city of Denver, Colorado.

The experiments show that it is feasible to predict the height for all buildings in the conterminous USA in under 6 minutes. Both the MLR and SVR method even accomplish it in under 30 seconds. The height prediction results show that the different prediction models struggle to accurately estimate the height for buildings in CBDs. The lowest achieved Mean Absolute Error (MAE) is 31.81m, whereas for the suburban and rural areas it is 1.41m. Adding additional, non-geometric features (e.g. census data) to the prediction models for one city (Denver) proved to be successful; the RFR method reduced its MAE from 1.35m to 0.96m for the suburbs, achieving sub-metre accuracy. The CBDs, however, are still problematic with an MAE of 16.87m.

These results show that for the suburban and rural areas, the accuracy recommendations from the CityGML specifications for LOD1 models can be met (5m limit). For the CBDs, improvement is required. The experiments also proved that the proposed methodology can be used to generate 3D city models of very large datasets if no elevation data is available. Moreover, the method is, in theory, generic enough to be applied outside the USA.

Acknowledgements

I would like to express my sincere gratitude to my first supervisor, Dr. Hugo Ledoux, for his guidance, valuable ideas and constant support during this graduation project. His positive mindset did help me greatly to stay optimistic during challenging times. My sincere thanks also goes to my second supervisor, Balázs Dukai, for his helpful suggestions, and his willingness to brainstorm for ideas and to assist with the technical difficulties I encountered. Finally, many thanks to Dr. Filip Biljecki and Dr. Jantien Stoter for their comments, to improve the quality of this thesis.

I would also like to thank my parents and my brother, for their infinite love and support during my entire studies. Lastly, I want to thank my close friends for always being there for me.

Contents

List of Figures

List of Tables

Acronyms

LOD	Level of Detail	1
OGC	Open Geospatial Consortium	1
VGI	Volunteered Geoinformation	1
LiDAR	Light Detection and Ranging	2
SRTM	Shuttle Radar Topography Mission	2
DEM	Digital Elevation Model	2
USA	United States of America	2
OCM	Open City Model	2
CBD	Central Business District	4
RFR	Random Forest Regression	4
SVR	Support Vector Regression	4
MLR	Multiple Linear Regression	4
XML	Extensible Markup Language	7
GML3	Geography Markup Language 3	7
JSON	JavaScript Object Notation	7
CRS	Coordinate Reference System	7
OSM	OpenStreetMap	8
RMSE	Root Mean Square Error	8
RF	Random Forest	10
MAE	Mean Absolute Error	10
SVM	Support Vector Machine	11
RFC	Random Forest Classifier	11
UBID	Unique Building Identifier	13
OLC	Open Location Code	13
UTM	Universal Transverse Mercator	14
SPCS	State Plane Coordinate System	14
DTM	Digital Terrain Model	15
DSM	Digital Surface Model	15
bagging	bootstrap aggregating	17
OOB	Out-of-Bag	18
OLS	Ordinary Least Squares	19
NPI	Normalised Perimeter Index	22
MBR	Minimum Bounding Rectangle	23
ACS	American Census Survey	24
MAPE	Mean Absolute Percentage Error	25
RMSPE	Root Mean Square Percentage Error	25
USGS	U.S. Geological Survey	31
AW3D30	ALOS World 3D - 30m	32
JAXA	Japan Aerospace Exploration Agency	32
GDAL	Geospatial Data Abstraction Library	33
MI	mutual information	40
VIF	Variance Inflation Factor	41
ECDF	Empirical Cumulative Distribution Function	57
DOI	Digital Object Identifier	85

1 Introduction

The demand for 3D spatial information — and particularly for 3D city models — has been increasing in recent years [Biljecki et al., 2015]. The extra dimension of the 3D data allows for a wider range of applications, which were not possible with only 2D data. Some of these applications are only based on the 3D geometries of the city models, while others also include semantic information or even add external data and domain-specific extensions [Ross, 2010]. Noise simulations, energy demand estimations, and visibility analysis are examples of possible applications (see Figure 1.1) [Stoter et al., 2008; Kaden and Kolbe, 2014; Lonergan and Hedley, 2016].

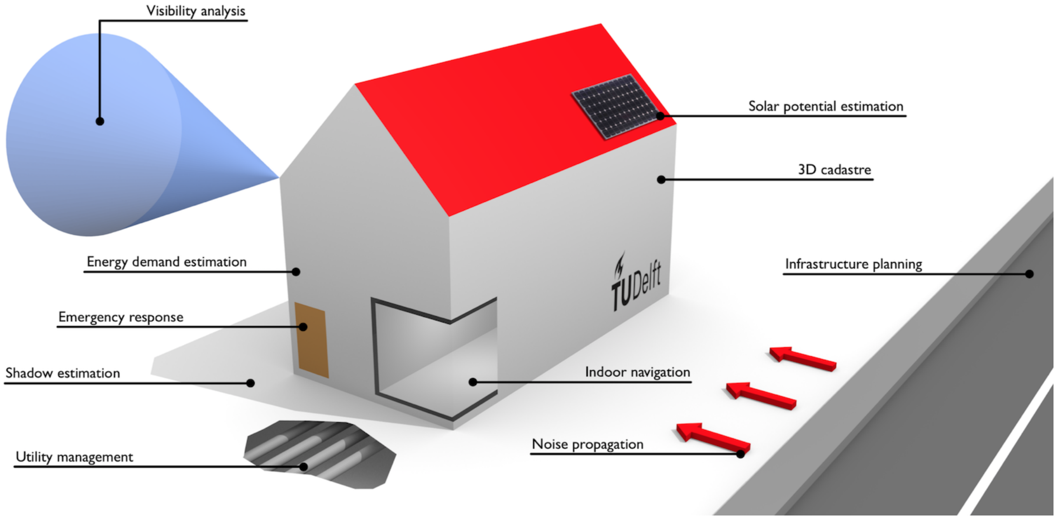


Figure 1.1: Examples of applications of 3D city models [Biljecki et al., 2015].

Different Levels of Detail (LODs) exist to represent the 3D city models. An increase in the LOD means an increase in both the geometric and semantic complexity [Gröger et al., 2012; Biljecki et al., 2016a]. The same object can be simultaneously represented in different LODs, which allows the user to choose the appropriate representation of a city depending on the application. The five different LODs as specified by the Open Geospatial Consortium (OGC) are (see Figure 1.2):

- LOD0: The 2D footprint of a building
- LOD1: A prismatic building (block model), often obtained by extruding the LOD0 model
- LOD2: A model containing simplified roof shapes and semantic classes for the boundary surfaces (e.g. wall, roof)
- LOD3: A model with architectural details including detailed wall and roof structures (e.g. doors, windows)
- LOD4: A complete model of the building, including its interior structure (e.g. rooms, stairs, furniture)

Several ways exist to create 3D city models. A popular and very simple and fast method is to use 2D building footprints in combination with building height data. The footprints are extruded to the specified height, creating an LOD1 model that can suffice for several applications [Stoter et al., 2008; Eeftens et al., 2013]. The 2D building footprints are often widely available as open data through government data-portals or as Volunteered Geoinformation (VGI) [Hecht et al., 2015]. However, the techniques to acquire the building heights — including Light Detection and Ranging (LiDAR) and photogrammetry — are usually time-consuming and expensive tasks. This seriously limits the availability of such datasets, and often the data is not available free of charge.

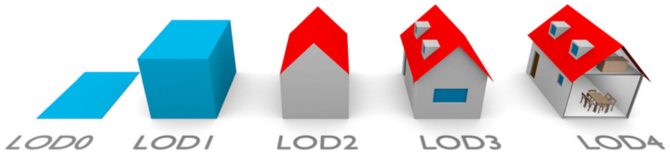
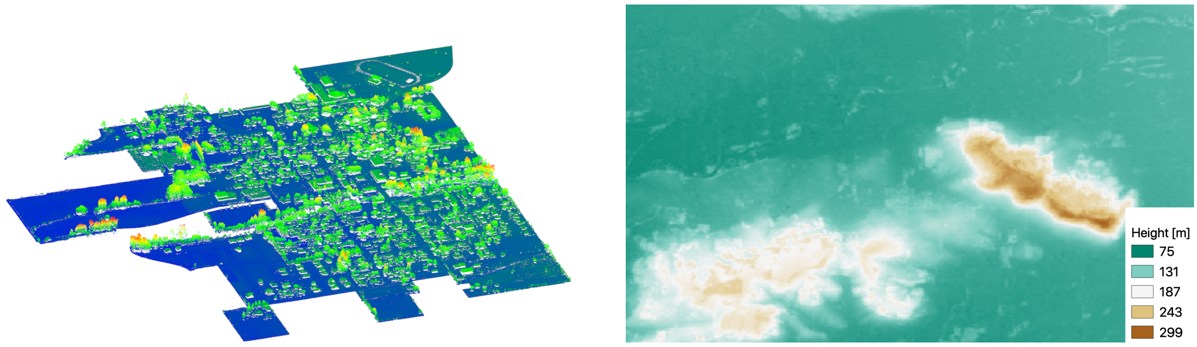


Figure 1.2: The five LODs as specified by the OGC for CityGML version 2.0 [Biljecki et al., 2016a].

The LiDAR datasets contain data points in 3D space (also known as *point clouds*, see Figure 1.3a), where each point can also contain extra information (e.g. RGB colour, classification). Point clouds are often only detailed enough to construct simple representations of buildings (i.e. LOD1 and LOD2), as insufficient information is present to model for instance chimneys or dormers. To compute the building height, the LiDAR data points that fall inside the 2D building footprint are used, and the footprint is then extruded to the computed height (see Figure 1.4). A problem that can arise with using LiDAR data is the 'mismatch' between the building footprints and the data points. Newer building footprints might only have ground points available, or they do have points available from a building previously situated at that same location.

Other data sources, such as the Shuttle Radar Topography Mission (SRTM), can also be considered for constructing simple LOD1 3D city models (see Figure 1.3b). This global Digital Elevation Model (DEM) with a resolution 30 metres is available free of charge [Smith and Sandwell, 2003]. However, the data's coarse resolution and low accuracy make it unsuitable for producing 3D city models. In many parts of the world — such as Africa — DEMs are often the only source of elevation data available, limiting the possibilities of generating 3D city models for these areas.



(a) LiDAR pointcloud

(b) DEM raster

Figure 1.3: Two different sources of elevation data for roughly the same area.

To overcome these problems, machine learning can be used. This thesis attempts to use such techniques to infer the height of buildings when no height data is available. Using these kinds of methods partly solves the data availability and fitness-for-purposes problems described above. Some experiments have been conducted in previous research. First; Biljecki et al. [2017] applied machine learning for height inference in the city of Rotterdam, the Netherlands based on building footprint characteristics and census data (see Section 2.3.1). Reference and training models are created from LiDAR data and 2D building footprints using the method described earlier (see Figure 1.4). The results are promising, making clear that machine learning can be used to bridge the gap if not enough height data is available.

Second, for the United States of America (USA), the Open City Model (OCM) (see Figure 1.5) is a nationwide 3D city model which, among others, makes use of a 2D building footprint dataset (`USBuildingFootprints`) created by Microsoft [BuildZero, 2019]. This footprint dataset contains 125,192,184 buildings, which are extracted from aerial imagery [Microsoft, 2018]. However, for the OCM the producers are not completely open about the techniques used to generate the building height results. A statement is made that the footprint area and its location are used in a simple regression analysis algorithm to estimate the building heights. Furthermore, they state that machine learning is only used in areas where no other height data is available, but it is not made clear beforehand for which areas this is the case. The lack of transparency about how machine learning is applied to this problem creates doubts about the results of the OCM.

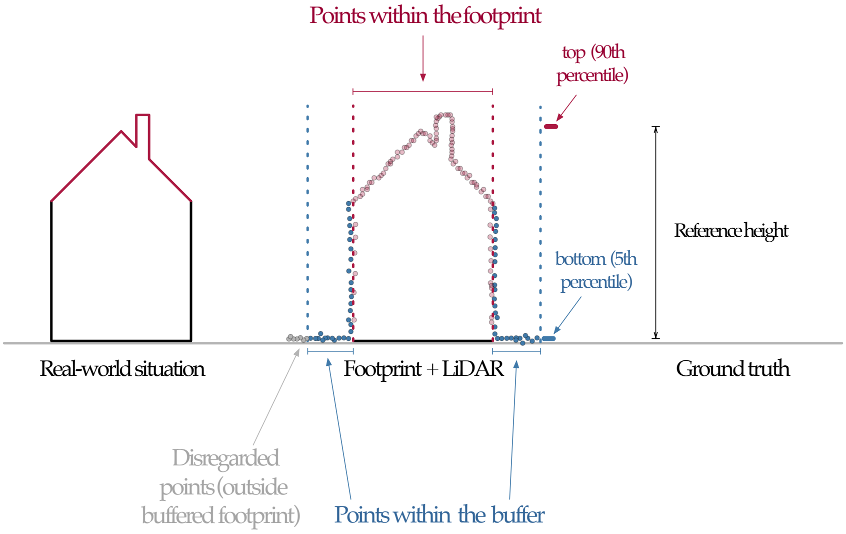


Figure 1.4: Computing building height from LiDAR data [Biljecki et al., 2017].

Also, the accuracy of the height estimations appears to be low (see Section 4.1 for a more in-depth analysis). The accuracy of the building heights must be as high as possible, making the 3D city models better suitable for applications requiring high-quality data. Therefore, the focus of this thesis will partly be on investigating different methods to improve on the OCM results using several machine learning techniques and building characteristics. Using cadastral data to possibly improve the prediction models — like in the research of Biljecki et al. [2017] — can be more challenging for the USA because the data is spread out over local governments. No national database comprising all relevant information about public and private parcels is available [Coalition of Geospatial Organizations, 2018].

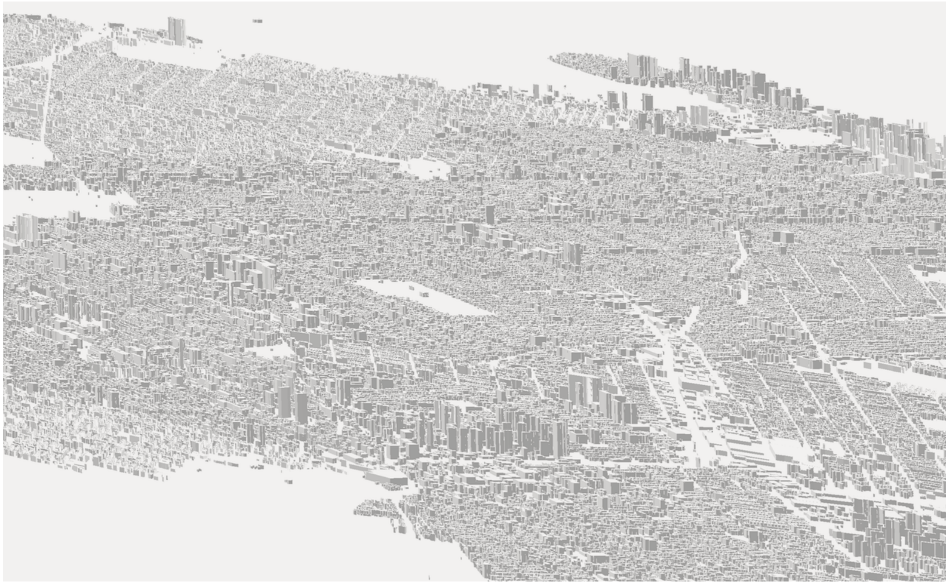


Figure 1.5: Example of the OCM data for the Kings neighbourhood in New York City, USA.

1.1 Objectives & Research Questions

The goal of this thesis is to develop a methodology to infer the building height for all building footprints in the conterminous USA — excluding Alaska and Hawaii — in the absence of height data, and to improve on the results of the OCM. The objective to improve means that the accuracy of the results should be considered to make the 3D city model(s) useful for analysis applications. The methodology should be generic enough to be applied outside of the USA, making it useful for other parts of the world where there is a scarcity of accurate enough elevation data. Lastly, the focus must be on scaling, which includes both the runtime of the algorithm and the different area morphologies that one can encounter (e.g. Central Business Districts (CBDs), suburbs, rural areas).

Based on the motivation and problem statement of this thesis, the main research question can be defined as follows:

Can the 125 million USA building footprints be assigned a height without making use of height data, and what accuracy can be achieved?

To answer the main question, the following five sub-questions are relevant;

1. What methods can be used to assess the accuracy of the building height estimations? And when are the estimations deemed accurate enough?

To perform the building height estimations with machine learning, the 2D building footprints must be characterised by certain *features*, which can be based on the geometries of the building footprints or the information from other data sources. These features provide information about the buildings, that in turn are used by the machine learning methods for learning. The geometric features are directly derived from the building footprints, while the non-geometric features are not. So;

2. What relations are present between the different geometric properties of the building footprints and the building height? And which subset is deemed 'optimal' for predicting building heights?
3. Are the geometric properties of the building footprints as training features sufficient for meeting the accuracy requirements?
4. What other features, besides the geometric properties of the building footprints, can be used in the machine learning algorithms to estimate building heights? And does including these features, even if they are incomplete, improve the accuracy of the estimations?
5. What methods can be used for scaling the machine learning techniques to the whole of the USA?

1.1.1 Scope

To define a clear research scope, the following remarks are made:

1. The focus will be on the building footprints of the conterminous USA, excluding the states of Alaska and Hawaii.
2. Three machine learning techniques are included; Random Forest Regression (RFR), Support Vector Regression (SVR), and Multiple Linear Regression (MLR). See Section 3.3 for in depth explanation of these methods.
3. The geometric features are extracted for all building footprints, while the non-geometric features will only be tested on a selected test area because they are not available everywhere (see Section 3.4.2 and 4.1).
4. The 3D city model(s) will be stored in the CityJSON format, excluding the CityGML format. It will be a block model; of LOD1. Section 2.1 explains the formats in more detail.

1.2 Thesis Outline

The content of this thesis is structured as follows;

In Chapter 2 a discussion on the state-of-the-art related to this project is provided. Particularly, important formats and standards for 3D city models are described, together with some examples of previous work of machine learning applied to 3D city models. Lastly, machine learning terminology and concepts that not all readers may be acquainted with are explained.

Chapter 3 presents the proposed methodology for meeting the objectives and answering the research questions. The different machine learning techniques and training features are discussed in more detail, together with how scaling can be applied and how the accuracy of the models is measured.

Further details on the implementation of the methodology, and the engineering choices made, are described in Chapter 4. The results of the implementation of the methodology are presented and analysed in Chapter 5. It includes the results of the three machine learning methods for the different test areas, and a comparison to the OCM.

Chapter 6 concludes the thesis. First, the degree to which the research questions have been fulfilled is reviewed, followed by the contributions to the state-of-the-art and a discussion about the implementation of the methodology. The discussion forms the basis for the recommendations for future work.

In addition to these main chapters, the following appendices are present;

Appendix A contains an evaluation of a reproducibility assessment for this research based on five different criteria, together with a self-reflection on the topic. Extra details on the datasets used in this thesis are provided in Appendix B. Appendix C shows extra plots about the feature analysis that did not fit in the main text. Lastly, Appendix D presents additional results for the three testing areas.

2 State-of-the-Art

In this chapter, the scientific research related to this graduation thesis is examined. In particular, a division between two main subjects is made: the formats and standards used for 3D city models — including characteristics and important factors to consider — and how machine learning can be used to generate or enrich these models. Also, the terminology necessary for understanding the methodology is reviewed.

2.1 Formats & Standards for 3D City Models

Different exchange formats exist for 3D models. Two popular formats for storing 3D *city* models in particular are CityGML and CityJSON [Gröger et al., 2012; Ledoux et al., 2019]. They share similarities, but also have some differences. In this thesis, I will focus on the CityGML data model and use the CityJSON encoding for the reasons described below.

CityGML is both an open data model and exchange format for 3D city models, and is standardised by the OGC [Gröger et al., 2012]. It is an Extensible Markup Language (XML)-based format, which makes use of Geography Markup Language 3 (GML3); an international standard for spatial data exchange. CityGML provides a common definition of the objects, attributes, and relations present in 3D city models. There is support for the most relevant topographic objects in cities, including their geometrical, topological, semantical, and appearance properties. The semantic classes are grouped in extension models, e.g. Bridge, Building and Vegetation.

CityGML does come with a drawback: its data files are often verbose and of a complex and hierarchical structure. Consequently, writing CityGML can be a difficult process, and the output is not suited for web applications. With this knowledge in mind, CityJSON was developed [Ledoux et al., 2019]. Unlike CityGML, CityJSON is not an official OGC standard. CityJSON provides a JavaScript Object Notation (JSON) encoding for the CityGML data model, which is easier to parse and allows for higher data compression than the XML-based format (6x in practice, see [Ledoux et al., 2019]). The hierarchies present in the CityGML schema are removed, making the data better understandable and human-readable.

Lastly, as discussed in Chapter 1, different LODs can be used for each of the semantic classes as defined by the OGC in the CityGML specification [Gröger et al., 2012]. While CityGML contains support for all five LODs, CityJSON lacks support for LOD4 [Ledoux et al., 2019]. It does support the so-called *TU Delft LODs* as defined by Biljecki et al. [2016a], which refine the first four LODs in CityGML for building objects. Furthermore, CityJSON only allows for one Coordinate Reference System (CRS) to be used for all objects, and this CRS should be an EPSG code[1]. These properties of CityJSON should not be an issue, because the focus in this thesis lies on buildings in LOD1 and a nation-wide CRS for the USA is used.

2.2 Building Height References in LOD1

When the geometry of a building is defined in LOD1, it can be ambiguous which part of the building structure is used to define the height of the roof surface [Biljecki et al., 2014]. CityGML does not standardise how the geometric reference of a building model should be stored, as there is no option to express the alternatives in the metadata. Additionally, for many datasets, it is often not known which

[1] European Petroleum Survey Group (EPSG) codes: `https://epsg.io`

geometric reference is used in practice. Figure 2.1 depicts the problem; seven different LOD1 block models can be derived from a building in LOD3, depending on the different height reference points. One can take into consideration constructions on the roof such as chimneys, leading in this case to a very tall building. If the tip of the roof is used, the LOD1 model of the building is already much lower. The deviation in height between the seven models is significant, which can have a great influence on further computations and analysis (e.g. volumetric computations).

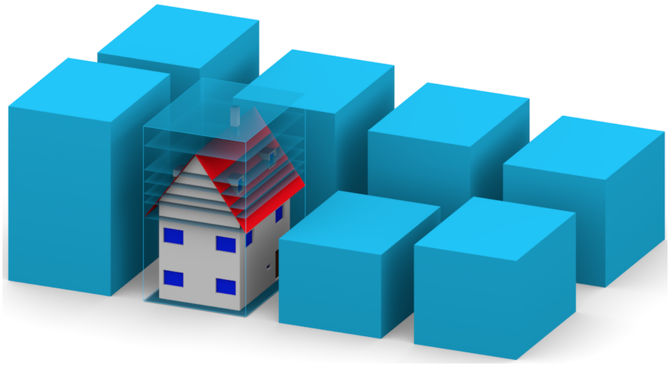


Figure 2.1: Seven different roof height references for a single building (in LOD3) [Biljecki et al., 2014].

Biljecki et al. [2014] also investigated common roof height reference points used in the production of LOD1 models. The following was found:

- *Airborne laser scanning and photogrammetry*: Often the median height of the points falling within the building footprint is used, matching to approximately half the height of the roof. One third or two-thirds of the roof height is sometimes also used.
- *Extrusion from footprints*: Attribute values such as the building height from OpenStreetMap (OSM), the number of floors, or the building height from cadastral records are used to extrude the building footprints. The data lineage is often unknown, causing uncertainty in the height of the block models.
- *Generalisation from finer LODs*: The bounding box around the building of a higher LOD is computed, which means the top surface can either be the roof tip or the highest point of a building (including features on the roof such as chimneys).

Not all representations of a building may be present in these three different techniques. With the number of floors, for example, objects on the roof are not considered. The reference points corresponding to those features are thus not present.

The selected roof reference point can also affect the Root Mean Square Error (RMSE) of the 3D city model. If the chosen reference point does not match the ground truth height measurements well, the RMSE will increase. This is an important consideration for this graduation thesis, since the LOD1 reference models and part of the training data are generated from LiDAR point clouds. The height reference that is used for the roof surfaces of the buildings is thus of high importance and can (significantly) impact the final results of the building height predictions.

2.3 Machine Learning for 3D City Models

3D city models cannot only be created through the means described in the previous section, but also for instance by using procedural modelling [Biljecki et al., 2016b] or by employing machine learning methods to estimate the height of the LOD1 buildings. The latter approach is seen as a subset of artificial intelligence, and it encompasses the study of computer algorithms that improve (i.e. *learn*) automatically over time by processing new experiences [Mitchell, 1997]. These algorithms create mathematical models based on samples (*training data*) to perform predictions or to make decisions with, without any human interaction [Bishop, 2006]. Several approaches exist, where the three main types of learning are 1) supervised learning, 2) unsupervised learning, and 3) reinforcement learning.

Supervised learning problems make use of training data that also contains the desired outputs; it has both *features* and *labels* available. Based on these training samples, inputs can be mapped to outputs. In *unsupervised learning* problems, the labels are not present in the input data. The learning algorithm has to find the structure in the input itself, e.g. discover similar groups of data (clustering) or determining the distribution of the data within the input space (density estimation). Figure 2.2 provides a visual example of what a supervised and unsupervised learning problem could look like. Lastly, a completely different type of learning is *reinforcement learning*, which deals with the problem of finding suitable actions to take in a given situation to maximise the reward [Sutton and Barto, 1998].

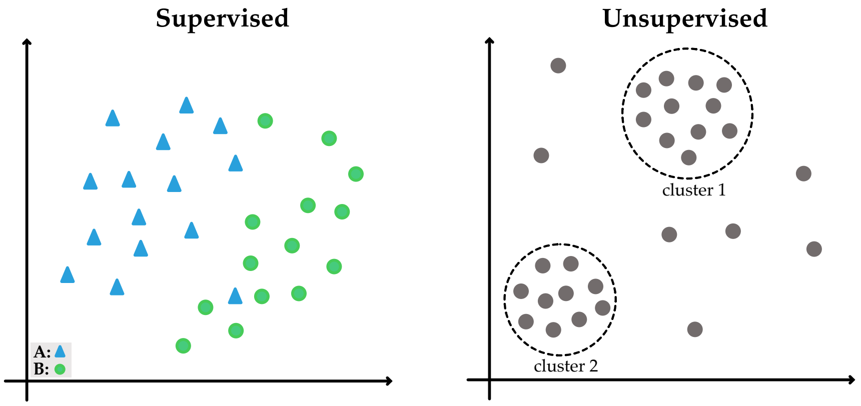


Figure 2.2: Comparison between supervised and unsupervised machine learning.

The focus of this thesis is on supervised learning problems, where the prediction models will be trained with both the features (i.e. characteristics of the building footprint) and labels (i.e. building height). Learning algorithms included in supervised learning are *classification* and *regression* (see Figure 2.3). Classification problems assign to each data point a discrete category out of a finite set. When the output of the algorithm is a numerical value within a given range, it is a regression problem. Height prediction for buildings falls in the last category of problems.

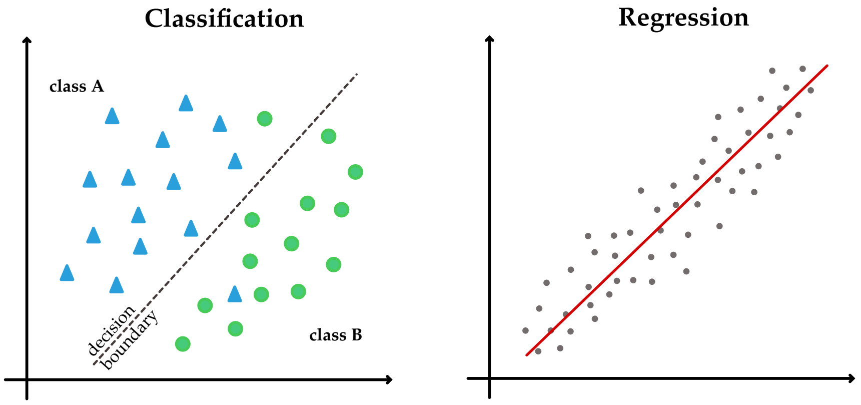


Figure 2.3: Comparison between classification- and regression-based supervised learning.

2.3.1 Building Height Inference

As discussed in Chapter 1, elevation data (e.g. LiDAR) is in theory essential for the construction of 3D city models. As these datasets are often unavailable, other solutions are sought to infer the height of buildings. Some more unconventional methods to do so include:

- *Sun ephemeris (shadows in satellite imagery)*: Shadows projected by buildings can provide an indication of their height [Dare, 2005; Shao et al., 2011; Comber et al., 2012; Liasis and Stavrou, 2016]. If the date and time of when the image was captured are known, the length of the shadow in combination with the altitude of the sun can be used to compute the building height. This method requires satellite imagery, making it vulnerable to the same problems as were present with LiDAR data: the data acquisition can be expensive and the data that is available might be outdated [Biljecki et al., 2017].

- *Attributes (number of storeys)*: 2D building datasets do sometimes contain information on the number of storeys. In combination with the storey height, the building height can be computed. Different practitioners use various storey heights, where values ranging from 2.8 to 3.5 metres have been found in different research papers [Goetz and Zipf, 2012; Guney et al., 2012]. These models are often not generated to be very accurate but are mostly used for visualisation purposes and interactive querying [Biljecki et al., 2017].

- *Local regulations / Zoning policies*: Municipalities often have local regulations or zoning policies describing the types of land use that are permitted or prohibited. These documents can also prescribe the maximum allowed building height for such zones. In areas with a high population density — and where land is scarce — these restrictions are often exploited as much as possible [Kontokosta, 2013]. It is, therefore, reasonable to assume that buildings in these zones are of the maximum permitted height.

Another way to infer the building heights is through machine learning techniques. Biljecki et al. [2017] describe a method using Random Forest Regression (RFR) to infer the building heights for 200,000 buildings in the city of Rotterdam, the Netherlands. They make use of three types of attributes (*features*) to train the prediction models. The first focuses on attributes from *cadastral data*, including the number of storeys, building usage, building age and the net internal area (usable floor area in units of the building). The second looks at *census data*, describing the demographics and socio-economic parameters of the 92 statistical neighbourhoods in Rotterdam. The features include the population density, average household size and average income. Lastly, *geometric attributes* such as the footprint area, shape complexity and the number of neighbouring buildings are derived for each 2D building footprint. The cadastral and census data are available through external data sources, while the geometric features are always available as they are derived from the building footprints themselves.

To cover a wide range of possible real-world scenarios, several prediction models are created based on different combinations of features. Based on the trees in the Random Forest (RF), the importance of the features is computed for each of the models. Features of very low importance can be eliminated, as they do not contribute much to the final predictions.

The method proposed by Biljecki et al. shows promising results when only geometric features are used. Using a ground truth model generated from LiDAR data and 2D building footprints (see Figure 1.4), the Mean Absolute Error (MAE) is computed for the different models. When only using the three geometric features described above, an MAE of 1.8 metres is achieved. It is found that the number of floors of a building is the most important feature, and combining it with other important features decreases the accuracy even more. The combination of the number of storeys, building age, and net internal area resulted in an MAE of 0.8 metres. However, none of these features can be directly derived from the 2D building footprints, limiting their application to the whole of the USA in this thesis project.

Anh et al. [2018] applied a similar approach to the city of Hanoi, Vietnam. The same geometric features as proposed by Biljecki et al. [2017] are used. The building usage is added as the only non-geometric feature, significantly limiting the total number of features used in the prediction model. Since no point cloud data is available for the study area, actual field surveys were conducted to obtain ground truth data for the building heights. To improve the model performance, cross-validation and grid-search techniques are used to adjust the hyperparameters of the model (see Section 4.3 for a further explanation). However, with an MAE of 7.12 metres, the performance of the prediction model is a lot less accurate than the one of Biljecki et al. [2017]. Even with the tuning of the model the error is more than two building storeys, which is especially significant for areas where the average building height is quite low.

2.3.2 Model Enrichment

Once a 3D city model is established, further computations can be performed to extract additional information to enrich the model. These predictions can also serve as some sort of quality control for the already existing attributes through verification.

Biljecki and Sindram [2017] describe a method that uses semantically rich LOD1 models to estimate building age (i.e. year of construction) using RFR. The prediction model is trained with geometric properties of the buildings and cadastral data, but a limitation arises in the fact that not every building in the study area has the complete information available. Precise estimations are difficult to obtain, but an approximate period (i.e. decade) of construction is achievable.

Henn et al. [2012] show that it is possible to automatically classify building types using a sophisticated classifier based on Support Vector Machines (SVMs). In this way, the 3D city model can be semantically enriched making it suitable for a wider range of applications. The SVM classifier is trained using a limited number of building characteristics and some attributes describing the spatial context the building is in (i.e. considering its neighbourhood).

Lastly, Biljecki and Dehbi [2019] propose a method to infer building roof types from a semantically enriched LOD1 model using a Random Forest Classifier (RFC), where an LOD2 model can be provided at no extra cost. Among others, the classifier is trained based on characteristics in the building footprints and their corresponding building height. The research showed that indicating whether a roof is flat or not can be achieved with high accuracy, but accurately classifying the actual roof type is more difficult.

So, 3D city models that are generated with machine learning techniques could also serve as input for other machine learning methods to enrich the models even further. The number, and complexity, of the features required for training the prediction models used in the enrichment process, depends greatly on the use-case. The 3D city models generated and enriched by machine learning can be used for various applications and different types of analysis provided that their accuracy is high enough.

2.4 Contributions

Given the current state-of-the-art, my contributions will be the following:

1. Investigate whether it is possible to apply machine learning on only the geometric features for the data of the USA. This includes deriving as many geometric features as possible from the 2D building footprints in an automated fashion. From these features, it must be tried to find an 'optimal' combination/ subset to perform the building height predictions with.
2. Scale the machine learning problem to a much larger extent than has previously been done. Not only should the 125 million footprints be processed, but these footprints are also in different types of areas. Distinctions between these areas morphologies will be made (i.e. CBD versus suburbs and rural), and differently trained prediction models will be applied.
3. Consider the different roof height references by providing a comparison between 3D city models generated with machine learning techniques and those generated from LiDAR data. This can provide insight into which height percentile fits the building height predictions the best.

3 Methodology

In this chapter, the methodology for addressing the research questions of this graduation thesis is described. Figure 3.1 shows the main steps. First, the actions required in the data preparation process are presented. Next, the scaling problem is addressed, followed by an in-depth analysis of the three different machine learning techniques. Extra focus is put on the training features that can be extracted from the 2D building footprints. In addition, different non-geometric features of possible interest to this learning problem are reviewed. Lastly, the different methods for assessing the accuracy of the building height predictions are considered.

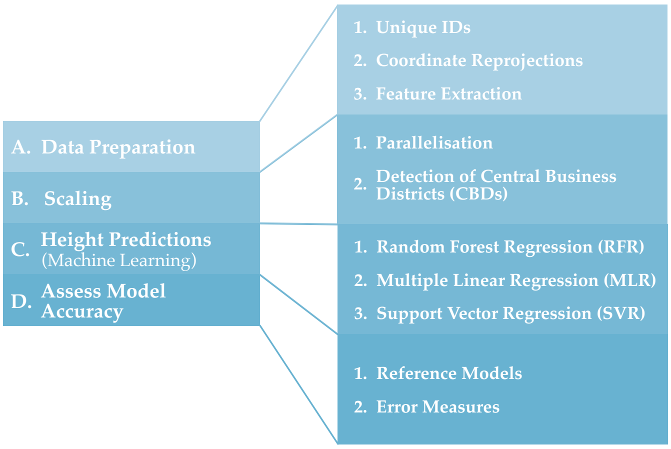

A. Data Preparation
B. Scaling
C. Height Predictions (Machine Learning)
D. Assess Model Accuracy
1. Unique IDs
2. Coordinate Reprojections
3. Feature Extraction
1. Parallelisation
2. Detection of Central Business Districts (CBDs)
1. Random Forest Regression (RFR)
2. Multiple Linear Regression (MLR)
3. Support Vector Regression (SVR)
1. Reference Models
2. Error Measures

Figure 3.1: The main steps in the methodology for addressing the research questions.

3.1 Data Preparation

The 2D building footprints, which are all available in vector format, require little pre-processing before the machine learning methods can be applied. Every building must have a unique identifier and be in the same Coordinate Reference System (CRS). Once these two steps are complete, the feature extraction is performed. The type of features are reviewed in Section 3.4, while the implementation is discussed in Section 4.2.2.

There are two different approaches for the building identifiers: 1) Unique Building Identifiers (UBIDs) or 2) state name abbreviations in combination with a unique number. The former uses a north axis-aligned bounding box and the Open Location Code (OLC) grid reference system designed by Google [Rinckes and Bung, 2019; Wang et al., 2019]. However, a problem with the UBID is that the OLC is based on longitude and latitude. All data must use a CRS supporting these units to compute the north axis-aligned bounding box. Since longitude and latitude are not ideal for the spatial operations required in the feature extraction process, a CRS with units in metres is preferred. Therefore, the second option for the unique identifiers is better suitable for this thesis project, because it does not rely on any CRS. It combines the abbreviation of the state name with a unique number; e.g. `NY_4351` for a building in the state New York. No look-up system is required to find out in which state the building is located because the identifier already provides this information.

The second pre-processing step focuses on the CRS for the building footprints. As discussed, the coordinates should preferably be in metres; Cartesian coordinates are preferred to longitude- and latitude-based CRSs. However, a combination of longitude, latitude and metres is used in the data (see Section 4.1), resulting in mixed units of degrees and metres. Several data viewers (e.g. `azul`[1]) and software packages do not support this. Besides, it can cause problems with spatial operations, such as computing the area of the building footprints. So, it is required that the coordinates are reprojected to a suitable CRS for the USA.

The first option for the coordinate reprojections uses the Universal Transverse Mercator (UTM) CRS, which divides Earth into sixty different zones, each of 6° of longitude in width [Langley, 1998]. Every location on Earth is assigned a zone and has a corresponding x- and y-coordinate in that plane. Because the USA is such a big country, it is part of many UTM zones (see Figure 3.2). A difficulty is that one state can be part of several zones. For example, the state of Texas covers UTM zones 13, 14 and 15. Reprojecting all building footprints at once is challenging. For each building footprint in a state dataset, it must be checked in which zone the building lies. Based on these findings, the data can be split into subsets to reproject them to their corresponding UTM zone CRS.

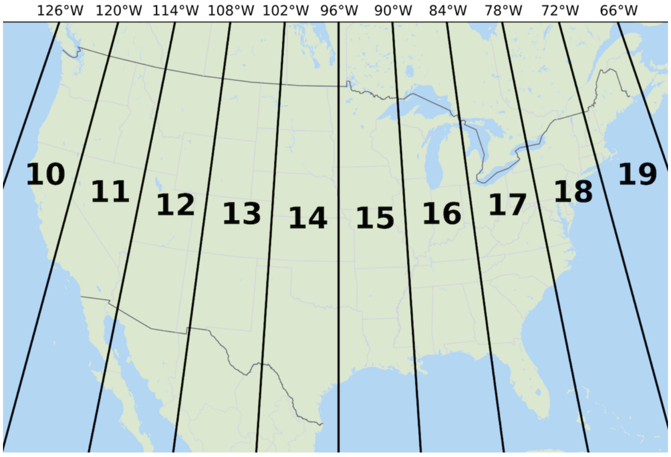


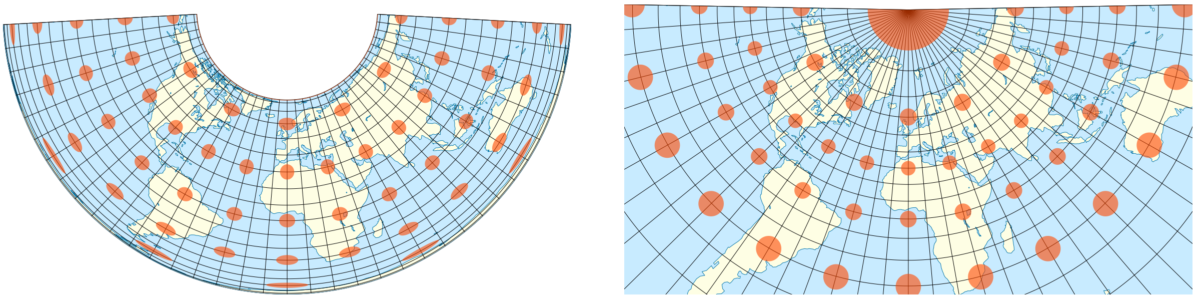
Figure 3.2: The different UTM zones for the USA. *Source:* Wikipedia.

A more user-friendly approach is to reproject all coordinates to a USA-wide CRS, provided that it minimises the distortion and that the coordinates are (or can be transformed into) Cartesian coordinates. Two options include the *Albers Equal Area Conic* and the *Lambert Conformal Conic* projections (see Figure 3.3). The former minimises the shape and linear scale distortion between the two standard parallels, while the latter portrays shapes more accurately than areas if located along middle latitudes [Kennedy and Kopp, 2000]. Both projections are commonly used for the conterminous USA. However, preserving the shape of the buildings is important in the feature extraction process, favouring the use of the *Lambert Conformal Conic* projection. The State Plane Coordinate System (SPCS), which divides each US state into six zones and uses Cartesian coordinates, also applies this projection for its mapping along the east-west axis [U.S. Geological Survey, 2017b].

3.2 Algorithm Scaling

The next step tackles the problem of scaling the algorithm to the whole of the USA. The problem is twofold. First, I must consider the different area morphologies the buildings are part of, such as Central Business Districts (CBDs) or more suburban and rural regions. Second, I have to analyse ways to minimise the runtime of the machine learning algorithms in terms of their training and height prediction stages.

[1] `https://github.com/tudelft3d/azul`



(a) Albers Equal Area Conic

(b) Lambert Conformal Conic

Figure 3.3: Two map projections suitable for the USA. *Source:* Wikipedia.

3.2.1 Detection of Central Business Districts

Buildings can be present in many different types of environments. I will focus on two area morphologies: CBDs and more rural and suburban regions. CBDs in the USA are often characterised by high-rise buildings densely located together, while the rural and suburban areas have generally lower buildings spread out over larger areas [Murphy, 1972]. Separating the two area morphologies before applying the machine learning algorithms seems like a sensible step to take. The morphologies can then be used in two different ways in the prediction model training process: 1) two separate prediction models are created, one for each area morphology or 2) a single prediction model is trained, and the area morphology is added as a feature in the training process. Results for the two approaches are presented in Section 5.2.

Separating the two area morphologies from each other requires elevation data. Directly utilising building height information on a nation-wide scale is not possible, hence why we try to predict them using machine learning. A solution to this problem is using both Digital Terrain Models (DTMs) and Digital Surface Models (DSMs), which can provide indications about the elevation in certain areas. DTMs provide a bare-earth raster representation of the Earth surface; i.e. no vegetation or man-made objects are present. DSMs do include objects and structures, such as buildings, on the terrain [Brovelli et al., 2004].

While a DTM or DSM on its own cannot provide information about the building heights, combining them can. Subtracting the DTM from the DSM results in a new raster dataset containing the heights relative to the Earth surface. Figure 3.4 illustrates what this looks like for the area around Manhattan in New York City. The heights in this new raster are not only for buildings but also for other objects such as vegetation. Excluding these objects from the data is difficult, especially if the data resolution is not high.

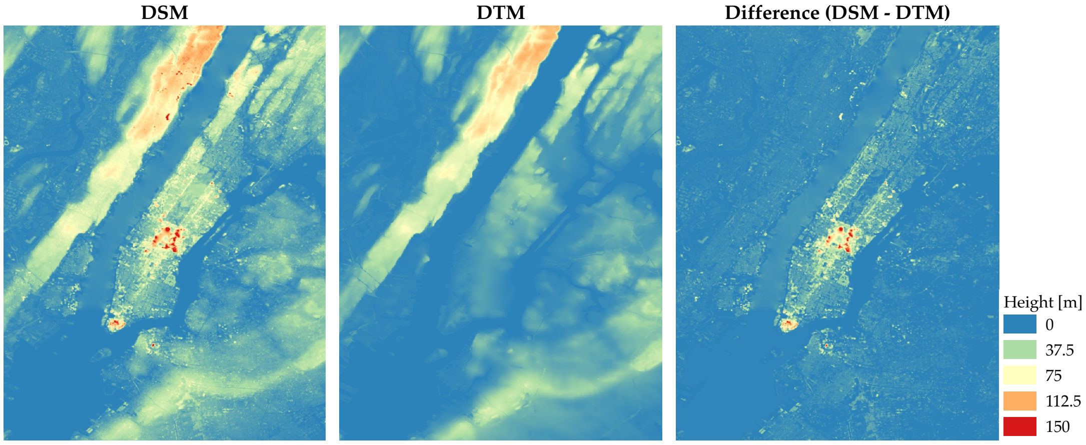


Figure 3.4: Comparison between a DSM and a DTM for the area around Manhattan, New York City. The difference between the two rasters provides the heights relative to the terrain.

The new raster dataset makes it easier to visually distinguish between regions with many high-rise buildings and the more suburban and rural areas. However, automatically detecting CBDs from such data poses several problems. First of all, the computation time will blow-up if the rasters are large and of high resolution. For a country like the USA, this is an important consideration to make. Secondly, clustering raster cells that belong to a CBD is not as straight forward as it might sound. It requires selecting thresholds to decide which cells have high enough values to be classified as a CBD. An in-depth analysis of the data is needed to do so. The next step in the process includes a vectorisation of the classification results to create the two different regions. However, a high variation in classifications between neighbouring cell results in many small polygons.

A second approach requires building footprint data. For each building, it must be checked how much of its area is covered by cells classified as a CBD. If more than x% is of the same area morphology, the building is assigned the corresponding class. However, it does again introduce a threshold problem and all buildings in a dataset are individually checked. For a large number of buildings, such in the USA, this becomes quite inefficient.

To simplify the problem, neighbourhoods can be introduced beforehand instead of detecting them in the data. Statistics are now derived based on these neighbourhoods and the underlying raster data. Performing these computations for the whole of the USA will still pose problems regarding computation time. However, only considering major cities reduces the complexity of the problem. It assumes that CBDs are only present in bigger cities, which in general is a valid assumption to make. A downside of this approach is that it requires an extra dataset, and such data is not always available.

In this thesis project, the latter approach with the extra neighbourhood dataset is used. Section 4.2.3 provides details on the implementation of the CBD detection for the USA.

3.2.2 Influencing the Algorithm Runtime

Several aspects can be of influence on the computation time of the machine learning methods. First of all, the complexity of machine learning methods differs. A simple linear regression model requires less computational power than creating multiple decision trees. Secondly, the *hyperparameters* of the machine learning methods influence the model training time. These values are set before the learning process starts, making them different from regular parameters that are derived through training [Claesen and de Moor, 2015]. The number of trees in a Random Forest (RF) is an example. An increment in the number of trees increases the complexity of the model and the computation time. Section 4.3 describes the hyperparameters for the different machine learning methods into more detail. Lastly, the number of features influences the training time of the prediction models. More features lead to a higher data complexity and a longer training period. An increase in the size of the training data has a similar effect.

These factors can only be influenced up to a certain extent. To further optimise the process, parallelisation of the training and predicting phases is possible. It allows running several jobs in parallel on different processors. In the case of a RF, the tasks are parallelised over the trees in the forest. Depending on the hardware, significant improvements are possible.

3.3 Machine Learning Methods

Before going into detail about the different features that can be useful in the model learning process, I will first focus on the three machine learning methods applied in this thesis. Their mathematical background, the assumptions they make, and their applicability to the height prediction problem are discussed.

3.3.1 Random Forest Regression

The first machine learning method I use for the building height predictions is Random Forest Regression (RFR). Breiman [2001] introduced RFs as part of the *ensemble learning* methods, which use several learning algorithms to obtain better predictive performance than was possible using these algorithms alone [Opitz and Maclin, 1999]. RFs have their foundation in decision tree learning: one can go from observations (represented in the branches) to target values (represented in the leaves). However, these decision trees are often grown deep, and they learn highly specific and irregular patterns present in the training set. It becomes difficult to predict values for unseen data, and this problem is known as *overfitting*. These kinds of models often have low bias and high variance, meaning that if the training set is split into two random sets, and decisions trees are fit to both parts, the results can be quite different for both trees [James et al., 2013]. RFs avoid this problem by averaging the multiple deep decision trees, where each tree is trained based on different parts of the same training set (see Figure 3.5). It slightly increases the bias and lowers the interpretability of the model, but at the same time, it boosts the model's prediction capabilities.

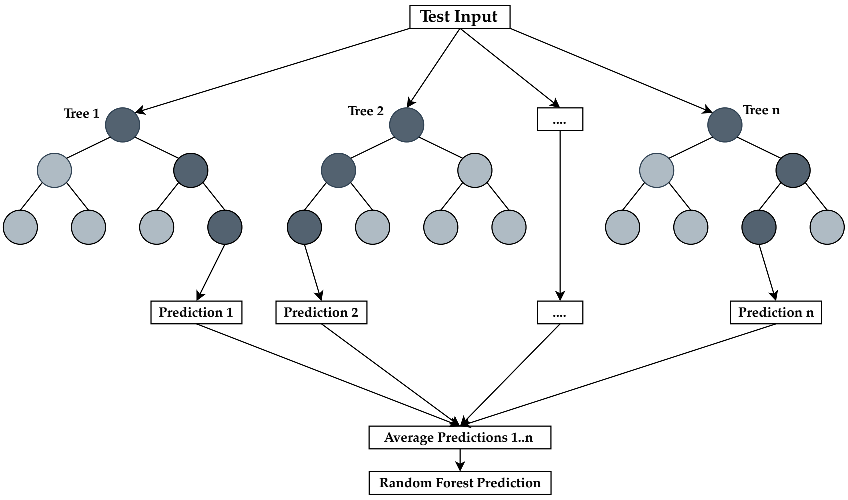


Figure 3.5: Decision trees that are generated by the RFR method for performing predictions.

A key concept in the RF solution is bootstrap aggregating (bagging), which provides a method to generate multiple versions of a predictor. An aggregated predictor then uses all these predictors to provide a final estimation [Breiman, 1996]. The goal is to improve the accuracy and stability of the machine learning algorithm by reducing the variance and helping to avoid overfitting. For a training set $X = x_1, \ldots, x_n$ with responses $Y = y_1, \ldots, y_n$ the following steps are performed T-times, with $t = 1, \ldots, T$:

1. Sample with replacement n examples from the training data $X, Y \rightarrow X_t, Y_t$
2. Train regression tree f_t on X_t, Y_t

The sampling with replacement means that the same sample can be present multiple times in this new training set. Then, predictions can be made by feeding an unseen sample x' to the individual regression trees, and averaging these outcomes:

$$\hat{f}_{avg}(x') = \frac{1}{T} \sum_{t=1}^{T} f_t(x') \tag{3.1}$$

The individual trees are highly sensitive to outliers in the training data. However, if the trees are uncorrelated to each other, then the average of the many trees is not. The *bootstrap* sampling makes this possible by creating the many different training sets.

Next, the bagging tree learning algorithm is slightly changed to create the RF. bagging allows features that are strongly correlated to the target variable to be present in many of the trees. As a result, the trees are correlated to each other as well. RFs solve this by randomly selecting features at each node in the tree, and using these features to split the node. It de-correlates the trees in the forest [Breiman, 2001].

The number of trees in the RF influences the *generalisation* error, which indicates how well an algorithm can predict the outcome values for previously unseen data [Breiman, 2001; Bousquet et al., 2004]. It uses the Out-of-Bag (OOB) error as an estimate of this generalisation error: for each training sample x_i, the mean prediction error is determined based on the trees that did not have x_i in their bootstrap sample [James et al., 2013]. Trees without the bootstrap sample are known as the OOB data, which normally is about one-third of the dataset, while two-thirds is included in the random subset. As the number of trees in the forest becomes larger, the generalisation error converges.

A strong point of RFs is their ability to compute the importance of the different features, which can be complex to calculate as it depends on the interaction with other variables [Liaw and Wiener, 2002]. First, a RF is fitted to the data and the OOB error is averaged over the entire forest for each data point. The importance of the j-th feature is computed at the end of the training session: the values of the feature are permuted in the training set, and the OOB error is determined based on this new data. Averaging the difference in the OOB error before and after the permutation over all trees provides a score, which is normalised by the standard deviation of the differences [Breiman, 2001]. Features scoring high are ranked higher than features scoring low. The feature ranking is useful for designing predictive models: only the important features are included, reducing the complexity of the prediction model [Grömping, 2009].

Another advantage of using RFs is that they do not make any assumptions about the data distribution. RFs are a non-parametric method, meaning they can handle both linear and non-linear relationships. The number of parameters in the model is data-dependent, making them different from dataset to dataset [Bousquet et al., 2004]. For the building height predictions, this characteristic can be of positive influence on the results. The relation between the features and the building height might not necessarily be linear (see Section 4.4). A RF will still be able to distinguish between the data in such cases.

3.3.2 Multiple Linear Regression

The second machine learning method I focus on, and the simplest of the three, is Multiple Linear Regression (MLR). It is based on linear regression, which models the relationship between a dependent variable (i.e. building height) and an independent variable (i.e. one of the features) [Bishop, 2006]. The *simple linear regression model* consists of the mean and variance functions [Weisberg, 2005; James et al., 2013]:

$$\begin{aligned} E(Y|X=x) &= \beta_0 + \beta_1 x \\ Var(Y|X=x) &= \sigma^2 \end{aligned} \tag{3.2}$$

where β_0 is the intercept and β_1 the slope (see Figure 3.6a). When the predictor is fixed at $X = x$, the expected value of the response is $E(Y|X = x)$. Similarly, $Var(Y|X = x)$ describes the variance of the response distribution. Varying the parameters in the functions allows for many possible straight lines. However, often the parameters are unknown and they must be estimated using data.

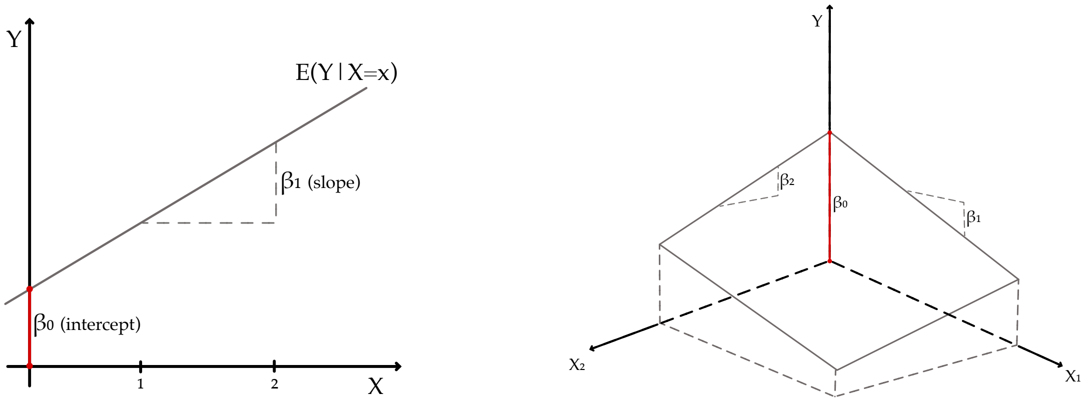
It is assumed that the variance is constant, where σ^2 is a positive value that is usually unknown. The observed value of the i-th response (y_i) is often not the same as the expected value $E(Y|X = x_i)$, hence the *statistical error* (ϵ_i) is introduced to describe this difference. It is defined as follows:

$$y_i = E(Y|X = x_i) + \epsilon_i \tag{3.3}$$

ϵ_i depends on the unknown parameters in the mean function, indicating that they are *random variables* and cannot be observed. It describes the vertical distance between the point y_i and the mean function $E(Y|X = x_i)$.

MLR is an extension of simple linear regression by allowing many terms in the mean function, instead of just one intercept and slope. In the case of n variables, the general form of the mean function with response Y and terms $X_0, \ldots, X_n$ is:

$$E(Y|X) = \beta_0 X_0 + \beta_1 X_1 + \ldots + \beta_n Xn \tag{3.4}$$



(a) Equation of a straight line

(b) Linear regression surface for 2 predictors

Figure 3.6: Characteristics of linear regression. Adapted from: [Weisberg, 2005].

where X_0 is always equal to 1, or the term is excluded if no intercept is present. The equation is also known as the linear function of the parameters: $n = 1$ represents a simple linear regression problem, $n = 2$ results in a three-dimensional plane (see Figure 3.6b), and $n > 2$ creates a hyperplane [Weisberg, 2005]. MLR's key idea is to add new variables to the function that explain parts of Y that are not explained by any of the other variables.

To fit the linear model and estimate the parameters of the model, Ordinary Least Squares (OLS) is used. It tries to minimise the residual sum of squares ($\epsilon_1^2 + \ldots + \epsilon_n^2$) between the observed values in the dataset and the predicted values in the linear approximation [James et al., 2013]. Figure 3.7 provides a visual representation, where the vertical lines represent the residuals (i.e. the signed value between the observation and fitted value).

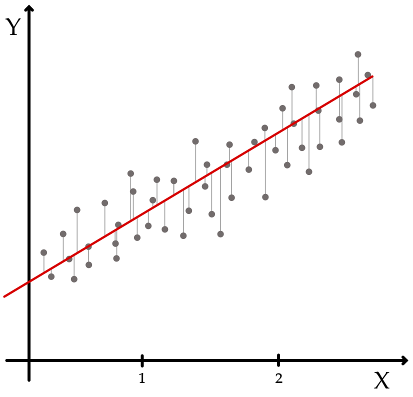


Figure 3.7: OLS fit of the data points, where the vertical lines are the residuals. Below the line the residuals are negative, while they are positive above the line.

To use MLR, several assumptions are made about the data and the error term:

1. *Linearity*: the relationship between the dependent and independent variables must be linear to make accurate estimations [James et al., 2013]. If the relationship is non-linear, the regression will under-estimate the true relationship.

2. *Homoscedasticity*: the variance in the errors is the same for different values of the response variable, regardless of the value of the predictor [Osborne and Waters, 2002]. If the residuals are randomly scattered around zero — showing a relatively even distribution — the ideal case is present. This means that $E(\epsilon_i | x_i) = 0$ [Weisberg, 2005].

3. *Independence of errors*: no information can be deduced for an error value from another error value because there is no relationship present between them [Weisberg, 2005]. So, the errors of the response value are uncorrelated.

4. *Normal distribution of errors*. When the sample size increases, this assumption for the residuals is not needed [Osborne and Waters, 2002].

5. *Lack of multicollinearity*: the predictors should have low inter-correlations among each other. If they express a strong relationship, the parameters of the model may be unreliable [Alin, 2010]. However, if the goal of the linear regression is prediction, multicollinearity is not the biggest problem, but it does affect the importance of the predictors [Hutcheson and Sofroniou, 1999].

These assumptions limit the applicability of MLR models. For the building height predictions, it is not guaranteed that the features express a linear relationship with the building height (see Section 4.4 for an in-depth analysis of the features). Fulfilling the other requirements can also be challenging.

3.3.3 Support Vector Regression

Support Vector Regression (SVR) is the last machine learning method I will focus on. SVR relies on kernel functions and, depending on the type of kernel, the model is either parametric or non-parametric. The use of non-linear kernels allows for more complex models to be fitted to the data. However, the complexity of the data fitting is more than quadratic with the number of samples, making it unsuitable for datasets with more than a couple of ten-thousand samples [TheKernelTrip, 2018]. For the USA, this introduces problems since the dataset contains roughly 125 million building footprints. An alternative is using linear kernels, which are faster but might not always fit the data.

SVR requires a loss function that includes a distance measure. Prior knowledge about the underlying distribution of the data is required to determine a suitable loss function [Gunn, 1998]. Vapnik [1995] designed the ϵ-insensitive loss function — an approximation of Huber's loss function [Huber, 1981] — that allows for robust estimates. For a linear ϵ-SVR, the goal is to find a function $f(x)$ that does not deviate more than ϵ from the target values y_i for all training data, and at the same time is as flat as possible [Smola and Schölkopf, 2004]. So, the errors are no problem as long as they are less than ϵ, but any deviation larger than this is not accepted. For a linear function f:

$$f(x) = (w \cdot x) + b \tag{3.5}$$

the *flatness* requirement is met for a small value of w, achieved by minimising the norm $||w||^2 = (w \cdot w)$. This convex optimisation problem can be formulated as follows:

$$\begin{aligned} &\text{minimise} \quad \frac{1}{2}||w||^2 \\ &\forall i : |y_i - (w_i \cdot x + b)| \leq \epsilon \end{aligned} \tag{3.6}$$

The assumption here is that such a function f exists, which can approximate all pairs (x_i, y_i) within ϵ precision. So, the optimisation problem should be feasible. However, this is not always the case, and allowing some errors is no problem. The *slack variables* ξ_i, ξ_i^* are introduced for each point to deal with the possibly infeasible constraint. All regression errors up to these bounds still satisfy the required conditions for the ϵ-SVR. The final formula to minimise becomes:

$$\frac{1}{2}||w||^2 + C \sum_{i=1}^{l} (\xi_i + \xi_i^*) \tag{3.7}$$

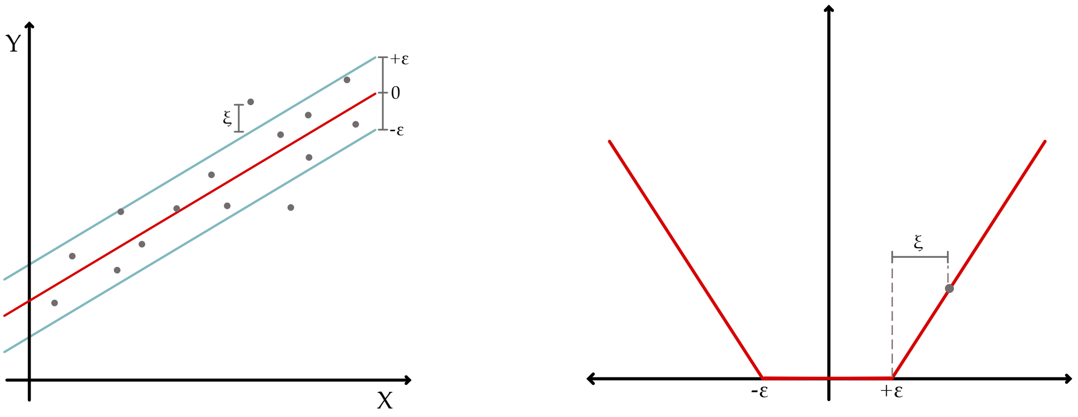
which is subject to:

$$\begin{aligned} &\forall i : y_i - (w_i \cdot x + b) \leq \epsilon + \xi_i \\ &\forall i : (w_i \cdot x + b) - y_i \leq \epsilon + \xi_i^* \\ &\forall i : \xi_i, \xi_i^* \geq 0 \end{aligned} \tag{3.8}$$

C is a constant (> 0) that determines the trade-off between the amount up to which deviations larger than ϵ are permitted and the flatness of f. All errors within ϵ distance from the observed value are ignored by the ϵ-insensitive loss function. The distance between the observation (y_i) and the ϵ-boundary provides a measure for the loss, which defines the ϵ-insensitive loss function $|\xi|_\epsilon$ as:

$$|\xi|_\epsilon = \begin{cases} 0 & \text{if } |\xi| \leq \epsilon \\ |\xi| - \epsilon & \text{otherwise} \end{cases} \tag{3.9}$$

Figure 3.8 provides a visual representation of the formulations. It shows the soft margin in the case of a linear problem, and the corresponding ϵ-insensitive linear loss function describing how the loss for a data point outside the ϵ-boundary is computed.



(a) Soft margin linear ϵ-SVR

(b) ϵ-insensitive linear loss function

Figure 3.8: The soft margin in the linear ϵ-SVR and the corresponding loss function. Adapted from: [Smola and Schölkopf, 2004].

The use of a linear kernel assumes that there is a linear relationship between the features and the building height. Similar to MLR, the performance of SVR is negatively affected by multicollinearity if there are large variations in the training data.

3.4 Training Features

To train the machine learning algorithms, it is required to have features that describe the building footprint characteristics. I differentiate between two types of features: geometric and non-geometric features. The former are derived from the 2D building footprints, while the latter are obtained from external sources. Section 4.4 discusses the influence of the geometric features on the building height to determine the optimal subset of features to perform the height predictions with.

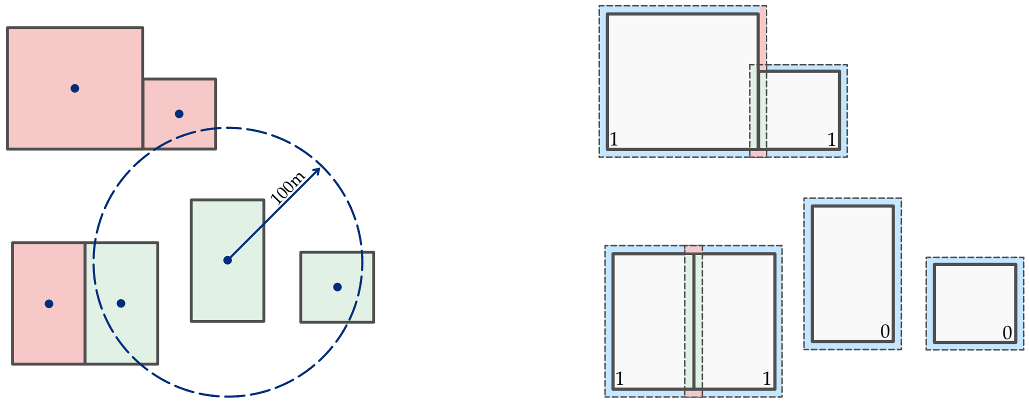
3.4.1 Geometric Features

Since the entire conterminous USA is considered, it is preferred to extract as many features as possible from the building footprints themselves. In this way, all footprints will have the same number of features available and the problem of missing data is avoided. In total, nine different geometric features are defined: *area*, *compactness*, *complexity*, *number of neighbours*, *number of adjacent buildings*, *length*, *width*, *slimness* and *number of vertices* (see Table 3.1). The footprint area, complexity (defined as compactness in this thesis) and the number of neighbours (features 1 to 3 in Table 3.1) are also implemented in the research of Biljecki et al. [2017]. I simply re-use these features as is. This thesis project includes extra features (features 4 to 9 in Table 3.1) to see whether it makes the prediction model better, resulting in more accurate building height estimations.

	Feature	Description	Computation
1.	Area	The area of the building footprint	-
2.	Compactness	The Normalised Perimeter Index (NPI)	$\frac{2\sqrt{\pi A}}{P}$
3.	Number of neighbours	Buildings within a range of 100 metres of the footprint	Centroid distance
4.	Complexity	The irregularities in the footprint	$\frac{P}{\sqrt[4]{A}}$
5.	Number of adjacent buildings	Buildings within 1 metre of the footprint	Buffer intersection
6.	Length	Longest edge of MBR	-
7.	Width	Shortest edge of MBR	-
8.	Slimness	Ratio of the sides	$\frac{F_{length}}{F_{width}}$
9.	Number of vertices	Total number of vertices in the footprint	-

Table 3.1: Features that can be derived from the 2D geometry of the building footprints.

The *area* of the building footprint is simply the surface that the building covers. It is used during the computation of two other features describing the boundary of the footprint: the compactness and complexity. The *compactness* is defined by the Normalised Perimeter Index (NPI), which uses the equal area circle and the perimeter of the polygon to describe how close the shape is to a circle. It is computed using $\frac{2\sqrt{\pi A}}{P}$, where A is the area and P the perimeter. A high NPI value means that the shape is closer to a circle than with a low NPI value. The normalisation makes the measure independent of the size of the polygon [Angel et al., 2010]. The *complexity* describes how many irregularities the footprint contains, and is computed as $\frac{P}{\sqrt[4]{A}}$, again with A as the area and P the perimeter. A high value indicates a smoother boundary, while lower values mean that the shape has a rough boundary [Sun et al., 2015]. These two features should be negatively correlated, as an increase in complexity often also means that the shape becomes less compact (see Section 4.4).



(a) Number of neighbours

(b) Adjacent buildings

Figure 3.9: The centroid method for computing the number of neighbours and the buffer method for computing the number of adjacent buildings. The number in the buildings in (b) indicates the number of adjacent buildings.

The *number of neighbours* can provide information about the type of environment the building footprint is located in. It is expected that in rural areas the number of neighbours is lower than in a city. For each 2D building footprint, its centroid is determined, and the number of other building centroids within its 100m radius defines the number of neighbours of a building (see Figure 3.9a). Based on my experiments, the distance of 100m provided the biggest contribution to the building height prediction compared to distances of 25, 50 and 75 metres. It must be noted that taking the centroid of a building with a large footprint area might affect the results because the distance from the centroid to the footprint edges is also bigger than for footprints with smaller footprint areas. However, computing the number of neighbours using a buffer method that takes into consideration all edges in the building footprint, is a lot more computationally expensive than the centroid method. Because the computation has to be performed for the roughly 125 million building footprints in the USA, the centroid approach is used in this thesis.

A similar measure to the number of neighbours is the *number of adjacent buildings*, which defines the direct neighbours of a building footprint (i.e. the footprints touching each other). As before, it is expected that this number is higher in cities than in rural areas, since for the latter buildings are more likely to be spread out over a larger area. The computation of this feature requires buffers: for each building footprint, a one-metre buffer is generated and intersected with nearby buildings (see Figure 3.9b). The number of intersections defines the number of adjacent buildings.

The next three features are derived from the oriented Minimum Bounding Rectangle (MBR) of the building footprint. A benefit of the oriented MBR is that it captures the shape of the building footprint better than a regular MBR, often called an axis-aligned bounding box (see Figure 3.10). The longest edge of the oriented MBR represents the *length* and the shortest edge represents the *width* of the footprint. Then, the *slimness* is computed as the ratio between the length and the width of the building footprint.

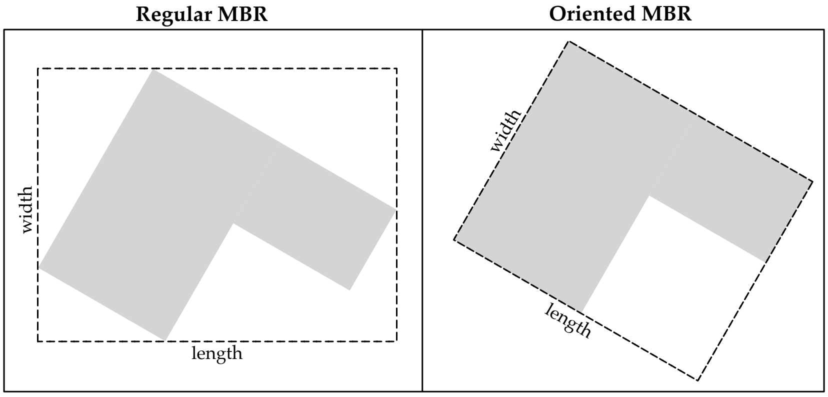


Figure 3.10: Comparison between a regular MBR and an oriented MBR.

Lastly, the *number of vertices* that make up the building footprint is counted. To avoid counting collinear points, the boundary of the footprints is first simplified using the Douglas-Peucker algorithm [Douglas and Peucker, 1973]. A line is drawn between the first and third point, and the distance between the middle point and the line is computed. If the distance satisfies the small threshold that is set, the collinear point is removed and the original shape of the building is preserved while the geometry is simplified (see Figure 3.11). The number of vertices can provide another indication of how complex the shape of the building is, where a higher number of vertices would mean a more complex shape.

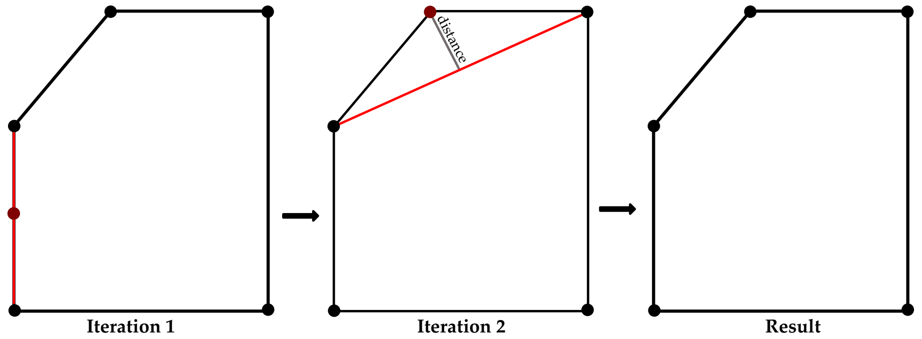


Figure 3.11: Example of the Douglas-Peucker algorithm. Vertices within a small distance are removed, such as in iteration 1. Vertices at a large distance are kept like in iteration 2. No other intermediate iterations are shown, only the final result.

3.4.2 Non-Geometric Features

Not only the geometric properties of the 2D building properties can say something about the building height, but also cadastral or statistical (census) data can provide additional information. The cadastral data can for example provide information on the *year of construction* (i.e. building age), the building *usage* or the *number of storeys* above ground, of which Biljecki et al. [2017] provides an analysis for the Netherlands. However, in the USA cadastral data is spread out over local governments and no national database comprising all relevant information about public and private parcels is available [Coalition of Geospatial Organizations, 2018]. This characteristic of the US national spatial data infrastructure makes it difficult to incorporate such information in the machine learning process.

Volunteered Geoinformation (VGI), such as OpenStreetMap (OSM), can possibly bridge this gap in information. The *building usage* can be classified under the building-tag[2], but it can also only simply say that the footprint does, in fact, belong to a building and no more information is provided. Another interesting attribute is *building:levels*, which indicates the *number of storeys* above ground[3]. However, the attribute data for building footprints are often incomplete, especially in the more rural areas where also fewer buildings are mapped [Hecht et al., 2013]. Furthermore, the buildings from OSM should be matched to the buildings in the USA dataset (i.e. `USBuildingFootprints`, see Section 4.1). An approach to solve this problem is the *centroid-based* method. It computes the centre of mass for all buildings and checks if the reference points lie within a building footprint in the other dataset. If this condition is true, there is a 'match' between the buildings. However, several factors are of influence during this process. First, there is not necessarily a correspondence between the OSM building footprints and the other dataset. Second, the centroid-method is affected by buildings represented as aggregated blocks instead of individual buildings. This degree of detailing can cause mismatches to occur. Applying this method to the entire USA can thus pose several difficulties.

The census data, on the other hand, is more widely available via the United States Census Bureau[4]. The USA use two types of surveys: the American Census Survey (ACS) and the decennial Census [ESRI, 2014; Bureau, 2020]. The former is performed every month every year, while the latter is only conducted once every ten years. The focus of the decennial census lies on counting the entire US population, while the ACS is more about the social and economic needs of the community. Data is distributed in different statistical entities, where I will focus on the *census tracts*. These are small and relatively permanent entities within counties, which are ought to be as homogeneous as possible concerning economic status, living conditions and population characteristics [Brown et al., 1994]. Because census tracts are used, only the five-year estimates of the ACS can be utilised. The one-year estimate does not provide enough information for analysing the smaller geographic areas.

From the two census data sources, the *population density*, *average household income* and *average household size* are available. This information can be assigned to the building footprints based on the census tract they are located in. The population density defines how many people live within one square mile of land for each census tract. Generally, a trend is present between the population density and the number of high-rise buildings: areas with a lot of high-rises are often much more densely populated than the more suburban areas [Cohen, 2015]. I try to exploit this trend to see if the feature can be used to infer the building heights in a given area (see Section 5.5). However, some drawbacks may be present. First, population density focuses on residential areas, while in the USA the CBDs are often characterised by tall buildings. Given these characteristics, the areas might not necessarily have a high population density. Second, the census tracts cover relatively large areas of land. Predicting the height for individual buildings is impossible because of this. Last, the diversity in buildings in the census tracts can also weaken the predictions, because both short and tall building can be present. In addition to the population density, the average household income and average household size are added for each tract to investigate if they can add more meaning to the models and improve the prediction results.

Besides cadastral and census data, information such as the number of amenities within a certain distance from a building can possibly also tell something about the building height. Rappaport [2008]

[2] https://taginfo.openstreetmap.org/keys/building

[3] https://wiki.openstreetmap.org/wiki/Key:building:levels

[4] https://www.census.gov/

describes that the number of amenities influences the population density in areas. Since these people need housing, the buildings in those areas are also possibly taller. Whether this is indeed the case is analysed in Section 5.5.

Lastly, one can think of features that directly provide a rough indication of the building height. An example that is tested in this thesis uses the heights that are present in a raster dataset, as described in Section 3.2.1. For each building, the cells covering the building footprints can be used to produce an average building height value. These values are then incorporated into the model training and predicting stages.

3.5 Measuring Model Accuracy

The last step in the methodology is about analysing the output of the three machine learning algorithms. A reference model with ground truth building height values is the first requirement to assess the accuracy of the building height estimations. I have to create the reference models myself because no reference data is available for the whole USA. Chapter 1 describes the method, where 2D building footprints are combined with LiDAR data. Each footprint is extruded to the computed height. However, creating these reference models does introduce uncertainty: different reference points can be used to define the roof surface (see Figure 2.1). Also, pre-existing reference models might use different roof reference points for the building height. Often it is unknown which reference point/ or height percentile is used.

Accounting for the different roof reference points in the pre-existing datasets is difficult. It requires additional information or data to determine the height reference of the dataset. LiDAR data allows for the computation of multiple heights for a given building footprint, each based on different height percentiles. Matching the building heights present in the dataset to the computed heights can give insight into the used roof reference point. However, it circles back to the original issue: an extra dataset is required to solve the problem, and addressing it is therefore difficult.

If a reference model is available, the quantitative accuracy of the output of the machine learning methods can be assessed with the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The MAE provides a risk metric that corresponds to the expected value of the absolute error loss [Chai and Draxler, 2014]. It is defined as:

$$\mathrm{MAE}(y, \hat{y}) = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} \tag{3.10}$$

where $\hat{y}_i$ is the predicted height for the i-th building, y_i the corresponding true value, and n the total number buildings the height prediction was performed for.

The RMSE includes the average magnitudes of the errors; the variance is penalised by giving more weight to errors with larger absolute values [Chai and Draxler, 2014]. This makes the metric sensitive to outliers, but also more suitable to analyse the model performance when model parameters are adjusted. The RMSE is defined as follows:

$$\mathrm{RMSE}(y, \hat{y}) = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}} \tag{3.11}$$

The values provided by these metrics might not be entirely meaningful. It can be more useful to represent the values as a percentage of the building height. An MAE of 2m has a lot higher impact in areas where the average building height is lower than in areas with a lot of high-rise buildings. The error in percentages can provide a better insight into the actual deviations for a given area. For this purpose the Mean Absolute Percentage Error (MAPE) and Root Mean Square Percentage Error (RMSPE) are introduced.

Lastly, the metrics aggregate all errors into one value. Cumulative errors can be used to provide more detail about the predictive models model(s) [Biljecki et al., 2017]. Based on the absolute height errors, the cumulative frequency is computed. It gives better insights into how many buildings have a height prediction within a certain error margin (e.g. 1m).

The qualitative assessment of the results is more difficult because there is no common agreement in terms of accuracy for LOD1 3D city models. We can generate such models with any height and deem them valid. The CityGML specification states that "In LOD1, the positional and height accuracy of points should be 5m or less, while all objects with a footprint of at least 6m by 6m should be considered" [Gröger et al., 2012]. However, this is not a requirement but a suggestion: "The accuracy requirements given in this standard are debatable and are to be considered as discussion proposals". Even though it is not a requirement, I will take this 5m recommendation into consideration during the analysis of the results.

4 Implementation

In this chapter, the implementation details of the methodology are discussed. The datasets that are used in this thesis project are presented and analysed first, followed by the programming specifics of the methodology. Next, the hyperparameter tuning problem for the RFR and SVR methods is discussed, where it is tried to find the best set of hyperparameters for the prediction models. At the end of this chapter, the suitability of the geometric features for predicting building heights is analysed.

4.1 Datasets

In this thesis project, various sorts of data are utilised. Table 4.1 provides an overview of the building footprint datasets that are used, including the LiDAR datasets for computing the building heights in case no height data is available in the footprints dataset. Appendix B contains the full information on where these datasets can be found, including their metadata.

Dataset	City	State	Purpose	#Buildings	#Buildings≥3m	Height data*
USBuildingFootprints	St.George	Utah	Training	26,996	26,658	Utah 1
	Cedar City	Utah	Training	8,846	8,160	Utah 2
	Wilson	Wyoming	Training	2,227	2,139	Wyoming 1&2
	Junction City	Oregon	Training	2,209	2,003	DOGAMI
	Hood River	Oregon	Training	2,685	2,461	DOGAMI
	Scio	Oregon	Training	428	396	DOGAMI
	Seattle	Washington	Testing	234	223	Washington 1
	Astoria	Oregon	Testing	3,794	3,723	DOGAMI
	Portland	Oregon	Testing	195,574	132,017	DOGAMI
3D Massing Toronto	Toronto	Ontario (Canada)	Training	420,852	413,114	-
Denver Building Outlines (2016)	Denver	Colorado	Additional Features	282,996	230,390	-
NYC Building Footprints	New York City	New York	Training	25,117	25,084	-

* See Appendix B for information about the LiDAR datasets

Table 4.1: The building footprint datasets used for training and testing the machine learning methods.

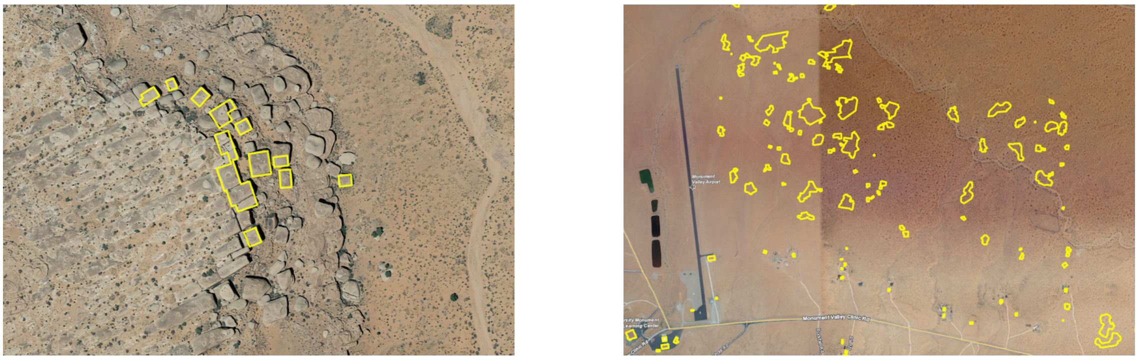
A fundamental data source are the 125,192,184 building footprints available in the USBuildingFootprints dataset [Microsoft, 2018]. These footprints are derived from aerial imagery and all use the EPSG:4326 CRS. The three machine learning methods will infer the building heights for these footprints based on their geometric features.

A visual inspection to assess the data quality of the USBuildingFootprints shows some mentionable findings. First of all, rectangular shaped rocks in desert areas are sometimes detected as buildings (see Figure 4.1a)[1]. It also occurs that other irregular shapes are detected when no identifiable features are present in the landscape (see Figure 4.1b). Similar artefacts are likely to be present at other locations.

Second, in the more built-up areas, the dataset often captures more building footprints than OSM. An example is shown in Figure 4.2a for Junction City, Oregon. The buildings shown in blue are present in OSM, and the buildings in red in the USBuildingFootprints dataset. The green rectangle marks the corresponding area of the satellite image. Only the buildings within the Junction City bounds are

[1] See the issue on GitHub: https://github.com/microsoft/USBuildingFootprints/issues/57

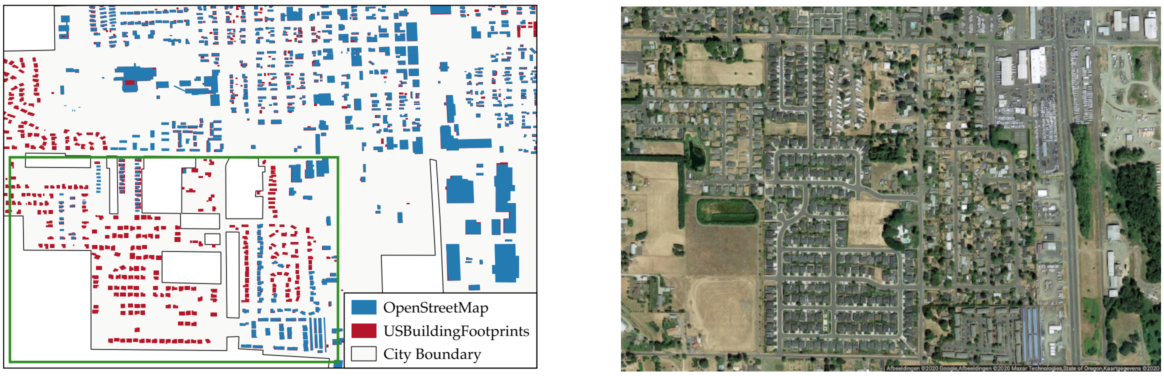
included in the comparison. In general, the buildings appear to be quite similar, but on the bottom left of the scene many buildings are only available in the USBuildingFootprints dataset. The satellite image in Figure 4.2b confirms that there are indeed buildings present on these locations. However, it is also clear that there are still buildings missing that are present in the real world.



(a) Rock formations detected as buildings

(b) Bare land detected as irregular building shapes

Figure 4.1: Irregularities in the USBuildingFootprints data in the Utah deserts. *Source*: GitHub.



(a)

(b)

Figure 4.2: Comparison between the coverage of USBuildingFootprints and OSM data for Junction City, Oregon. (a) shows that the former contains more buildings and the satellite image in (b) confirms that these buildings do exist. The green rectangle marks the coverage of the satellite image.

The last aspect to consider for the USBuildingFootprints is how accurately the building footprints are represented. In cases that many buildings are built close together (e.g. terraced houses) the buildings seem to be aggregated into bigger blocks. The reasoning for this is that the footprints are detected in areal imagery and automatically polygonised, making it difficult to distinguish between separate buildings if they are attached. The problem is illustrated in Figure 4.3 for part of Manhattan, New York City. On the left the building footprints as provided by the New York City open data portal[2] are shown, while on the right the data from USBuildingFootprints is presented. The data in the USBuildingFootprints dataset is clearly overly simplified. This will influence the geometric features that are derived from the footprints.

From the USBuildingFootprints I derive six training datasets for smaller cities and rural areas (see Table 4.1). The building footprints are extracted from the corresponding state dataset using a polygon that contains the city bounds (see Appendix B). The extent of each of these datasets is shown in Figure 4.4. These footprints are then combined with LiDAR data using 3dfier[3], where the 10th percentile is used for

[2] NYC open data portal: https://opendata.cityofnewyork.us

[3] 3dfier tool developed by the TU Delft 3D Geoinformation Group: https://github.com/tudelft3d/3dfier
Make use of the --stat_RMSE and --CSV-BUILDINGS-MULTIPLE options for different height percentiles with RMSE information

ground points, and the 90th percentile for the roof points. The difference between these two heights provides the building height of the footprint. The 90th height percentile is selected because the 50th and 75th height percentiles both included many buildings with very low building heights (<1m). This allows for bigger training datasets than when the lower height percentiles are used. An analysis of the influence of different height percentiles is provided in Section 5.7. The point clouds that are used to compute the building heights are pre-processed using `LAStools`[4], including clipping them to the city bounds extent, converting height values from feet to metres, and reclassifying the points into the unclassified, ground, vegetation and building classes in case the original data did not have any clear classification already.

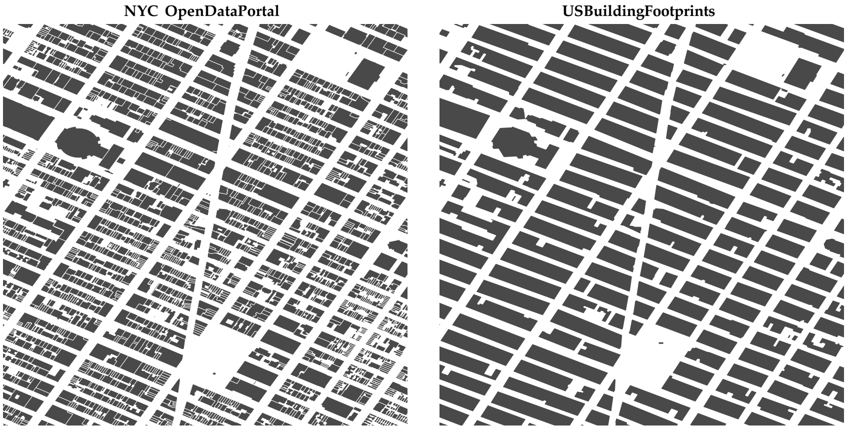


Figure 4.3: Building footprints in the `USBuildingFootprints` dataset are highly aggregated into bigger blocks compared to another local data source. The area shown is part of Manhattan, New York City.

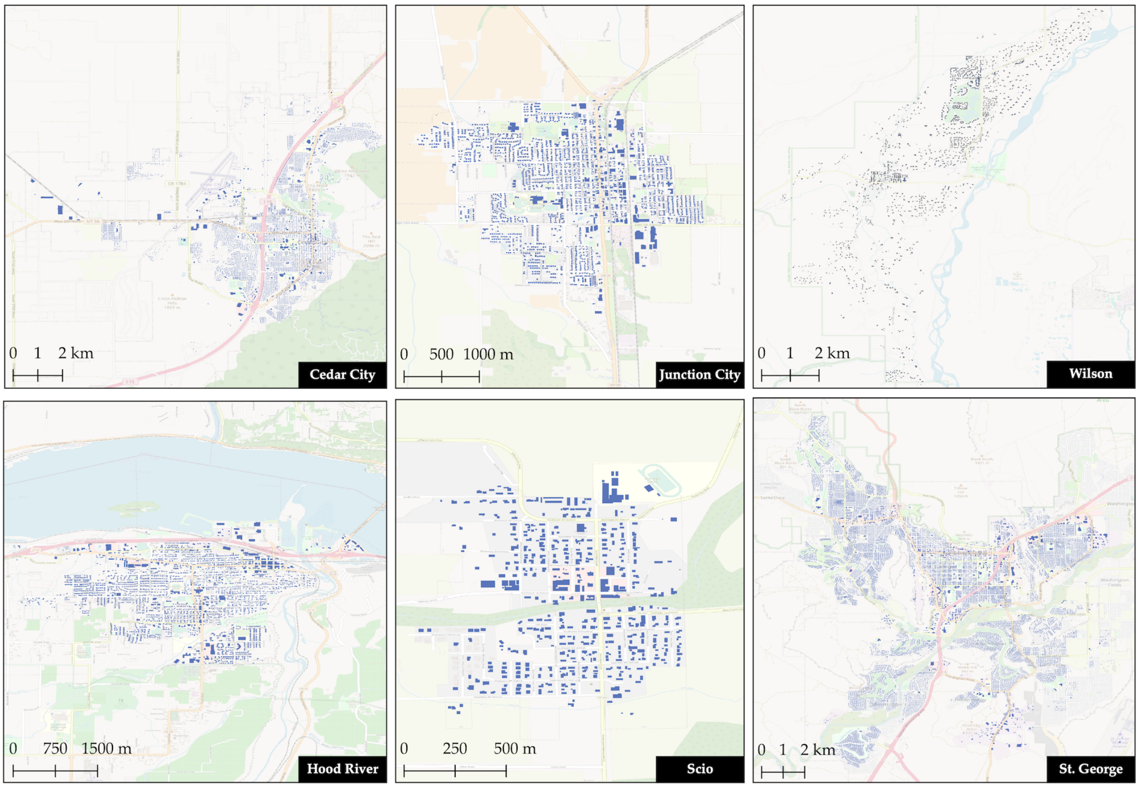
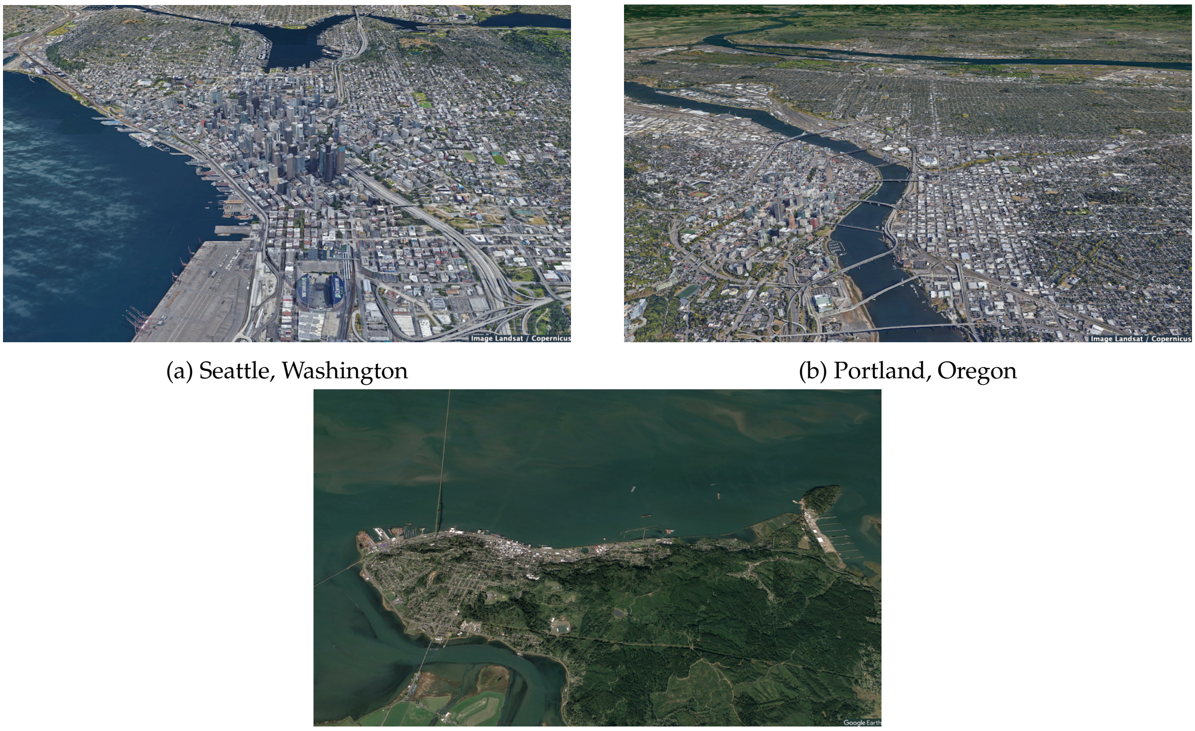


Figure 4.4: The six suburban and rural datasets that are extracted from the `USBuildingFootprints` data and used for training the prediction networks.

[4] `LAStools` software: `https://rapidlasso.com/LAStools/`

Similarly, three testing datasets are created, each serving a different purpose. Seattle is included for its CBD to inspect the accuracy of the height predictions in such areas. Portland does not have any CBD detected, making it interesting to see how this affects the height predictions in areas with tall buildings. Lastly, Astoria is included as a more rural area. Figure 4.5 shows the urban layout of these three different areas.



(a) Seattle, Washington

(b) Portland, Oregon

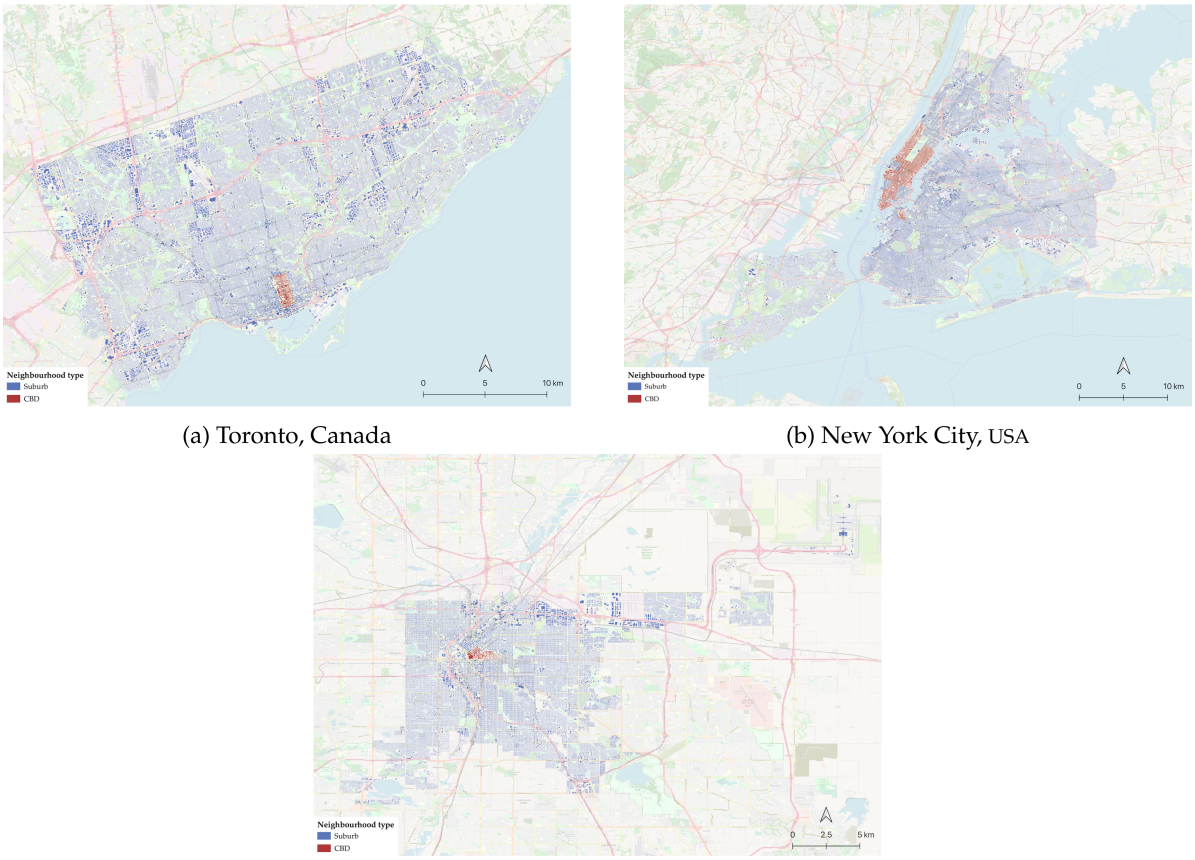
(c) Astoria, Oregon

Figure 4.5: Google Earth snapshots showing the city and town layout of the three different test areas. The Seattle CBD contains many tall buildings, while Portland has relatively many short buildings. Astoria is a more rural area when compared to the other two.

In addition to the `USBuildingFootprints` data, I use three other building footprint datasets which are already enriched with building height information. The first is the city of Toronto, Canada which contains 420,852 buildings [City Planning Toronto, 2019]. Toronto is not in the USA, but it lies close to the American border and the city shows similarities in its layout to other cities in the USA. The dataset is selected because of the limited availability of US data that is enriched with height values. The building heights in the Toronto dataset are obtained from LiDAR, site plan information, 3D models and oblique aerial imagery. The extent of the dataset is shown in Figure 4.6a, where the blue buildings are classified as suburbs, and the red buildings as the CBD.

The second dataset is the city of New York [NYC OpenData, 2020], and I include this dataset because of its many tall buildings. Toronto only has a small CBD, so by including NYC, I expand the training set for this type of area making the predictor model more versatile. The dataset contains a total of 1,085,008 building footprints, but only the 25,117 buildings in the detected CBD are considered (see the red buildings in Figure 4.6b). The building heights are either measured from controlled terrestrial image sources or taken from the Department of Buildings plan diagrams [Kamptner, 2020].

The third dataset is Denver, Colorado [Denver Regional Council of Governments, 2019]. This dataset is used for the trial with the non-geometric features. The full dataset contains 319,749 buildings, but I only consider the 282,996 buildings inside the census tracts (see Figure 4.6c). The dataset does contain building heights, but it states that these heights are obtained through visual inspection. Since the accuracy of these heights is unclear, I will not incorporate the data during the training process for the whole of the USA. For comparison reasons between the model with geometric features and the one enriched with the non-geometric features, the uncertainty in building height is no problem because all results will be based on the same dataset. An advantage of this dataset is that it contains the building type for each building. Therefore, no extra cadastral data has to be collected. For the average household income, the ACS with the five-year average from 2014-2018 is used. The average household size is obtained from the 2010 Census. Lastly, the population density data is found in an ACS dataset for the years 2013-2017. The time range differs for each dataset, but the census is only provided every ten years, and for the ACS data a one-year difference exists because no similar period could be found. However, if possible, the most up-to-date data should be used. Details about these datasets, including their sources, are included in Appendix B. Figure 4.7 shows the distribution of the three census-related features over the census tracts in Denver.



(a) Toronto, Canada

(b) New York City, USA

(c) Denver, USA

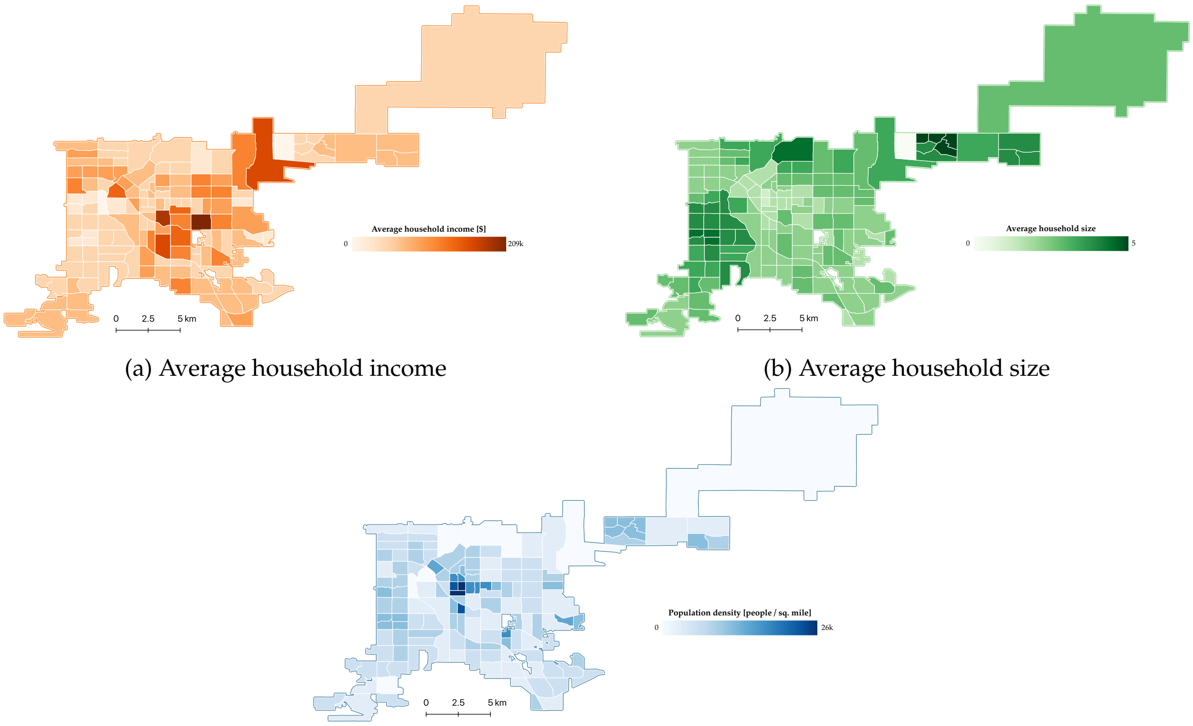
Figure 4.6: Building footprint datasets. Buildings in blue are detected as suburbs, while the buildings in red belong to the CBD.

In addition to these census datasets, OpenStreetMap (OSM) data[5] is used to extract the number of amenities within a 250m radius of a building. The data is filtered on the `fclass` attribute and only the following amenities are kept: `supermarket`, `pub`, `post_office`, `cafe`, `museum`, `convenience`, `bank`, `restaurant`, `department_store`, `school`, `hospital`, `kindergarten`, `library`, `theatre`, `pharmacy`, `bar`, `cinema`, and `kiosk`. These are amenities that are often present in city centres, where the population density and building heights are generally higher.

To complete the training set, data is required for detecting CBDs to enable scaling of the algorithm. First of all, data describing the neighbourhoods in the major cities of the USA is needed. The `Zillow - US Neighbourhoods` dataset is used for this purpose. It includes a total of 17,300 neighbourhood boundaries [Zillow, 2018]. Secondly, elevation data is required to compute the heights relative to the terrain. The DTMs are obtained via the U.S. Geological Survey (USGS), who provides raster data at several resolutions

[5] OSM data Colorado: `https://download.geofabrik.de/north-america/us/colorado.html`

[Archuleta et al., 2017]. I select the 1 arc-second dataset (approximately 30m resolution), as it provides coverage for the conterminous USA [U.S. Geological Survey, 2017a] and it has the same resolution as the DSM datasets. The DSMs are part of the ALOS World 3D - 30m (AW3D30) dataset developed by the Japan Aerospace Exploration Agency (JAXA) [JAXA, 2020], which provides a global coverage. Both data sources use metres for the vertical extent. A side note that must be made is that it is possible that there is some mismatch between the two datasets, because the models were not captured at exactly the same time. For the city of Denver, the DTMs and DSMs are used to compute the rough building height estimates that are used as an extra non-geometric feature.


Average household income [$]
0
209k
0
2.5
5 km
Average household size
0
5
0
2.5
5 km
Population density [people / sq. mile]
0
26k
0
2.5
5 km

(a) Average household income

(b) Average household size

(c) Population density

Figure 4.7: Distribution of the three statistical features over the census tracts for the city of Denver, Colorado.

Lastly, the Open City Model (OCM) data (see the end of the introduction in Chapter 1) is required to perform a building height comparison with my machine learning models. The OCM is the only other model — that I know of — that generates building heights for the USA. These height estimations are combined with areas that already have measured heights available, meaning that not all building heights in the USA are generated using machine learning techniques [BuildZero, 2019]. An analysis of the data for the city of San Diego, California, showed that many buildings are assigned similar height values. Out of the 160,000 footprints that were inspected, almost 54,000 buildings have a height of 5.73m; one-third of the total amount of data. The city centre shows another trend: buildings with larger footprint areas are all assigned the maximum height (9.31m) that was found in the dataset. In reality, the city centre has multiple tall buildings and thus these height estimations do not seem an accurate representation of the real-word scenario.

The 3D city models of the OCM are distributed in both CityGML and CityJSON using the `EPSG:4979` CRS (i.e. mixed units of degrees and metres are present). These models are not always geometrically valid[6]. Most of the buildings have a wrong orientation of the shells; the outer shell should have its normals point outwards, while the interior shell should have the normals point inwards (i.e. right-hand rule). Other errors that occurred include consecutive points that are the same in a ring (i.e. repeated points), self-intersections, and non-manifold cases where the shell is not simple.

[6] Geometric validity is checked with `val3dity`: `https://github.com/tudelft3d/val3dity`

The 3D buildings from the OCM are converted to 2D footprints to make the data usable for this research. For each building, the ground surface is extracted from the CityJSON model. The 2D footprints can be used in a database to perform spatial operations on and to detect the buildings that fall within the city boundary of the three selected test areas (see Table 4.1). In total, three different county datasets from the OCM are used: Clatsop County for Astoria, Multnomah County for Portland, and part of King County for Seattle. See Appendix B for the specific dataset names.

4.2 Programming Specifics

In this section, the technical details of the implementation are discussed[7]. The methodology is implemented in `Python`, where `psycopg2` [Gregorio and Varrazzo, 2020] and `scikit-learn` [Pedregosa et al., 2011] are the two main libraries on which several parts of the implementation rely. The machine learning operations are performed on a Linux server running `Ubuntu 18.04.4 LTS`. It contains forty `Intel(R) Xeon(R) CPU E5-2650 v3 @ 2.30GHz` CPUs, 126Gb of physcial RAM and 32Gb of swap memory.

4.2.1 Data Preparation

Pre-processing of the `USBuildingFootprints` dataset is performed in both `Python` and in a `PostgreSQL` database extended with `PostGIS` [Strobl, 2008]. First, the unique identifiers are added to the features in the `GeoJSON` data files. Each state dataset is read, and the state name is converted to the state abbreviation via a look-up in a dictionary. A unique number is added for each feature, and once all features are enriched with the unique identifier, the contents are dumped into a new `GeoJSON` file.

Secondly, this data is loaded into the database using the `ogr2ogr` tool from the Geospatial Data Abstraction Library (GDAL) software library [Warmerdam et al., 2019]. For each state, the data is stored in a separate table, making the data more manageable. The coordinate reprojections to the *Lambert Conformal Conic* projection are performed in the database. It requires adding the `ESRI:102004` (USA Contiguous Lambert Conformal Conic)[8] CRS to the database because this projection is not present by default. The coordinate reprojection is then performed using the `ST_Transform()` function in `PostGIS`. The new geometries are stored in a new column added to the existing table. For these operations it is important that the table is *unlogged*, which significantly improves the write performance because no data is written to the write-ahead log [The PostgreSQL Global Development Group, 2020]. However, these tables are not persistent: if the database crashes or is shut down, the data is lost. Storing the final result in a *logged* table is therefore essential because all geometric features are extracted based on these geometries. Losing the geometries after a crash or shut-down means the whole process must be repeated again, which is time-consuming because of the large number of building footprints. The coordinate reprojections are integrated into the feature extraction workflow described below.

4.2.2 Extracting Training Features

All geometric features are extracted in the `PostgreSQL` database using functions available in `PostGIS v2.5.x` or higher. The connection with the database is set-up through `Python` and the `psycopg2` library. Once the connection is established, I iteratively loop through all tables in the database. For each state table, an unlogged temporary table is created with the same column structure as the original table. All operations are performed on this new table: for each feature, a new column is added and the table is filled using an `UPDATE` statement based on a given query. The nine geometric features are computed sequentially, and once all features are extracted the data is copied to a logged table. The unlogged table is removed to save storage space.

[7] Source code available on GitHub: `https://github.com/ImkeLansky/USA-BuildingHeightInference`

[8] USA Contiguous Lambert Conformal Conic: `http://epsg.io/102004`

4.2.3 Scaling: Detecting Central Business Districts

The biggest task associated with scaling the algorithm to the whole of the USA is the CBD detection. First, the DTMs and DSMs for the areas corresponding to the ones present in the `Zillow` neighbourhoods dataset are pre-processed. The DTMs from the USGS make use of `EPSG:4269` (NAD83), while the AW3D30 DSMs are in `EPSG:4326` (WGS84). `gdalwarp` is applied to reproject the USGS rasters to `EPSG:4326`, but this results in re-sampling of the data using nearest neighbour interpolation. Because of this, the values in the raster do slightly change. For the AW3D30 DSMs, the -9999 values (of void pixels) are changed to `nodata` using `gdal_translate`. This step is necessary because of the subtraction of the DTMs from the DSMs.

Next, the raster tiles covering the neighbourhood areas are merged into one big raster dataset using `gdal_merge`, making it easier to subtract the DTM from the DSM in the following step. Once the difference between the raster cells is calculated, zonal statistics are computed for each neighbourhood using `QGIS` [QGIS Development Team, 2020]. Each neighbourhood is enriched with the mean, median, minimum and maximum height values, including the variance and standard deviation of the cell values inside the neighbourhood. It is feasible to run the computations in `QGIS` because I only calculate these statistics for the 17,300 neighbourhoods and not for the entire USA.

Extracting the CBDs is performed using a filtering method based on the zonal statistics. Both the median and mean were used in the trials, but from a visual inspection, the latter seemed to provide better results. In addition to the mean, the maximum height value is used to set a minimum height that must be present in the neighbourhood. Some neighbourhoods express a high mean value because they contain many cells with similar height values. These neighbourhoods are then detected as CBDs, even though there are no tall buildings present. By applying this extra constraint, false positives are removed (see Figure 4.8).

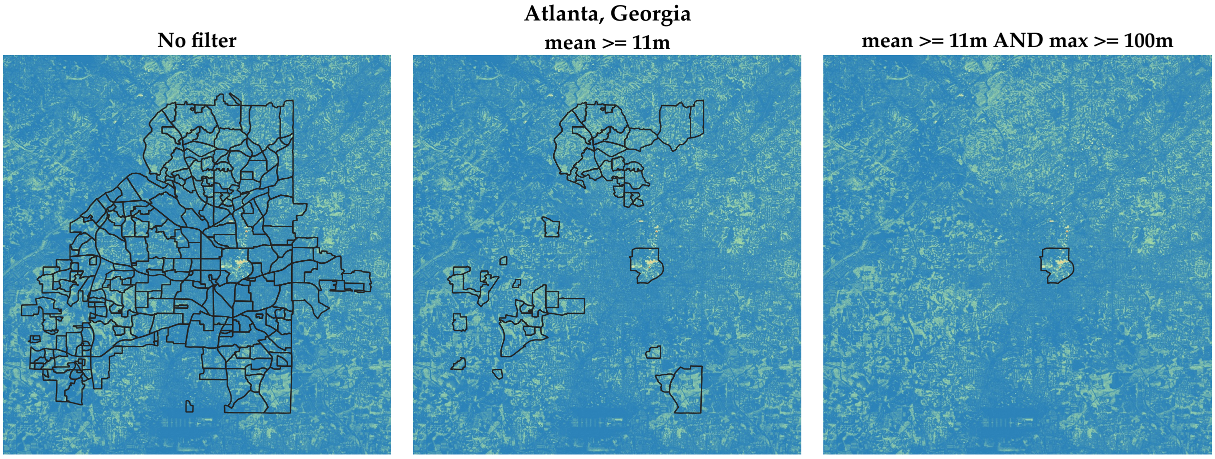


Figure 4.8: Example of CBD detection for Atlanta, Georgia. Only filtering on the mean height value in a neighbourhood results in many false positives. Adding the minimum height constraint eliminates these cases.

Several combinations of thresholds were tested, but — based on a visual inspection — the combination of $(mean \geq 11\text{m}) \wedge (maximum \geq 100\text{m})$ provided the best results. The selected neighbourhoods are imported into the `PostgreSQL` database, where for each building it is checked whether it falls inside a CBD or not.

This method does fail to detect certain CBDs, often because the neighbourhood is too large or too large height differences are present in a single neighbourhood. Also, the results are highly dependent on the thresholds that are set. For different size neighbourhoods or different types of environments, other threshold values might work better.

4.2.4 Machine Learning Methods

The implementation of the three machine learning methods heavily relies on the `scikit-learn` library. All data is directly read from the database and temporarily stored in a `pandas DataFrame`, separating the CBD data from the rural and suburban data. In the training stage, only buildings with a height greater or equal to 3m are retrieved (roughly one building storey). Next, the features and labels (i.e. building height) are separated from each other. Because most of the features are on different scales, I first apply *feature scaling* to all numerical features. The range of the features is normalised, removing the chance for certain features to dominate the prediction. The features are standardised by removing the mean and scaling to unit variance:

$$x' = \frac{x - \bar{x}}{\sigma} \tag{4.1}$$

where x' is the normalised value for the feature, $\bar{x}$ the mean of the feature vector, and σ the standard deviation. Feature scaling is more important for SVR and MLR than for RFR. However, the latter does provide more meaningful feature importance values if the features are on the same scale. Therefore, feature scaling is applied before running any of the machine learning methods.

ID	bld_type
1	Residential
2	Commercial
3	Industrial

(a) Original

ID	bld_type_residential	bld_type_commercial	bld_type_industrial
1	1	0	0
2	0	1	0
3	0	0	1

(b) One-hot-encoded

Table 4.2: Example of one-hot-encoding for some values of the building type feature.

In the case we are also dealing with non-numerical (i.e. categorical) feature values — such as the building types in the Denver dataset — *one-hot-encoding* is applied before training the prediction networks. The feature values are used to create additional features describing the type of the building footprint. All of these columns have a value of 0, except the column that contains the actual value of the original categorical feature, which has a value of 1 (see Table 4.2). Introducing one-hot-encoding increases the complexity of the model because of the extra columns, which can be problematic when many categorical features are present or when a feature has many different values.

The next step includes training the prediction models. Once fully trained, the model can be stored to disk. When the model is needed, it can simply be loaded from the file and be directly used to predict building heights; no re-training is required. All features on which the predictions are performed are scaled first as described above. The prediction results are linked to the building identifier in a list of tuples, which are later used for storing the results in the database.

4.3 Hyperparameter Tuning

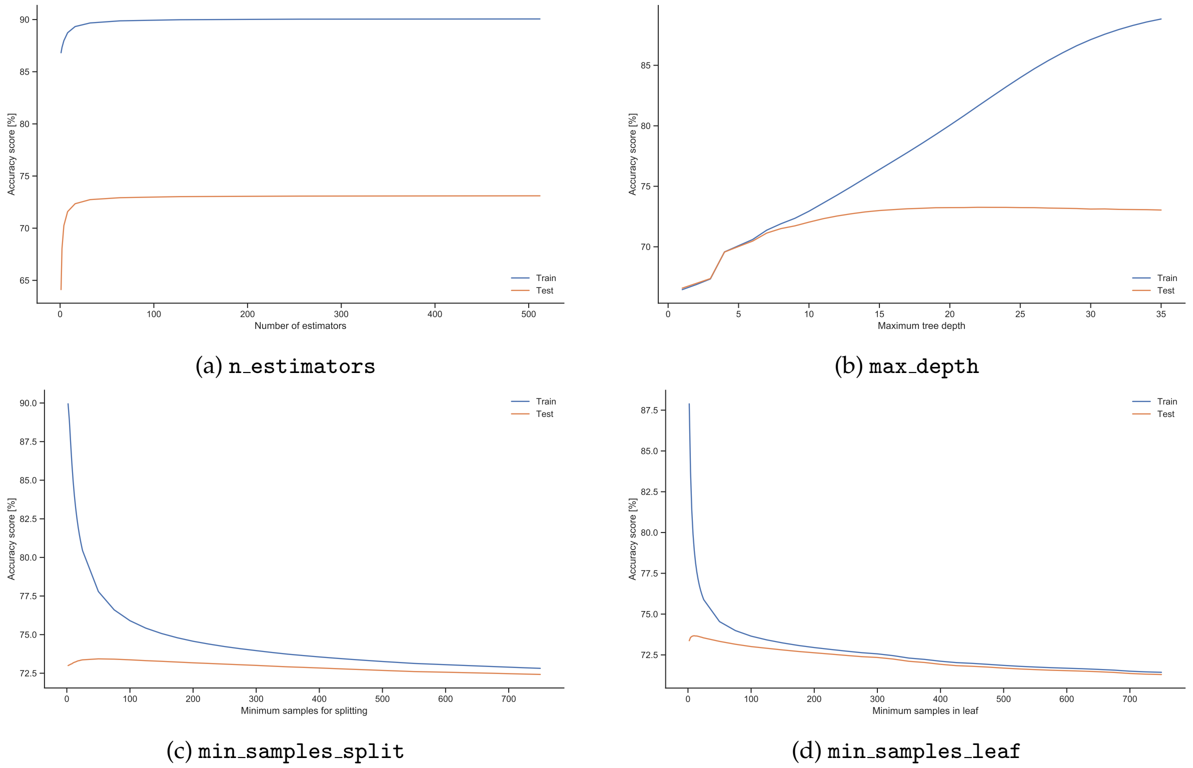
Before running the machine learning algorithms to generate final results, the hyperparameters of the models must be tuned. This is done to find the optimal set of hyperparameters for a machine learning method. Depending on the learning algorithm, different hyperparameters can be tuned. In this thesis, only the hyperparameters of the RFR and SVR methods require tuning because MLR is a very basic approach with only a few parameters. Table 4.3 and 4.4 show the hyperparameters — as available in the `scikit-learn` library — for RFR and SVR respectively. For each method, a grid with possible values for the hyperparameters is defined. To get an initial idea of which values to pick, each hyperparameter is isolated and altered to check its influence on the model performance. To measure the performance, the MAPE is converted to the mean absolute percent accuracy by computing *100 - MAPE*. The accuracy is computed for both a training and testing set. Both subsets contain 50% of the full training dataset (see Table 4.1) and they do not include any overlap in buildings. Figure 4.9 and 4.10 show the influence of the different hyperparameters on the RFR and SVR models respectively. For these graphs, only the rural and suburban data is included.

Hyperparameter	Description
`n_estimators`	The number of trees in the random forest
`max_depth`	The maximum tree depth
`min_samples_split`	The minimum number of samples required in an internal node
`min_samples_leaf`	The minimum number of samples required in a leaf node
`max_features`	The number of features to consider when splitting a node
`bootstrap`	Whether to enable bootstrap samples for building the tree or not

Table 4.3: The hyperparameters for the RFR in the `scikit-learn` library.

In RFR, increasing the *number of estimators* generally leads to better predictions because the model can generalise better. However, the complexity of the model also grows. At some point, no significant improvements in the model performance are present any more (see Figure 4.9a). I do not select an unnecessarily high number of trees to reduce computational complexity.

The *maximum tree depth* limits the depth to which the tree can grow. Figure 4.9b shows that an increase of the tree depth improves the performance of the model on the training set. However, the performance of the test set stagnates at some point and even starts to slowly drop as the value increases. If the tree depth becomes too high, the model will start overfitting because it adjusts the model to fit specific patterns in the training set.



(a) `n_estimators`

(b) `max_depth`

(c) `min_samples_split`

(d) `min_samples_leaf`

Figure 4.9: Influence of the different RFR hyperparameters on the model performance for a training and testing set. These graphs are generated based on the rural and suburban data. 50% of the data is used for training, and the other 50% is used for testing.

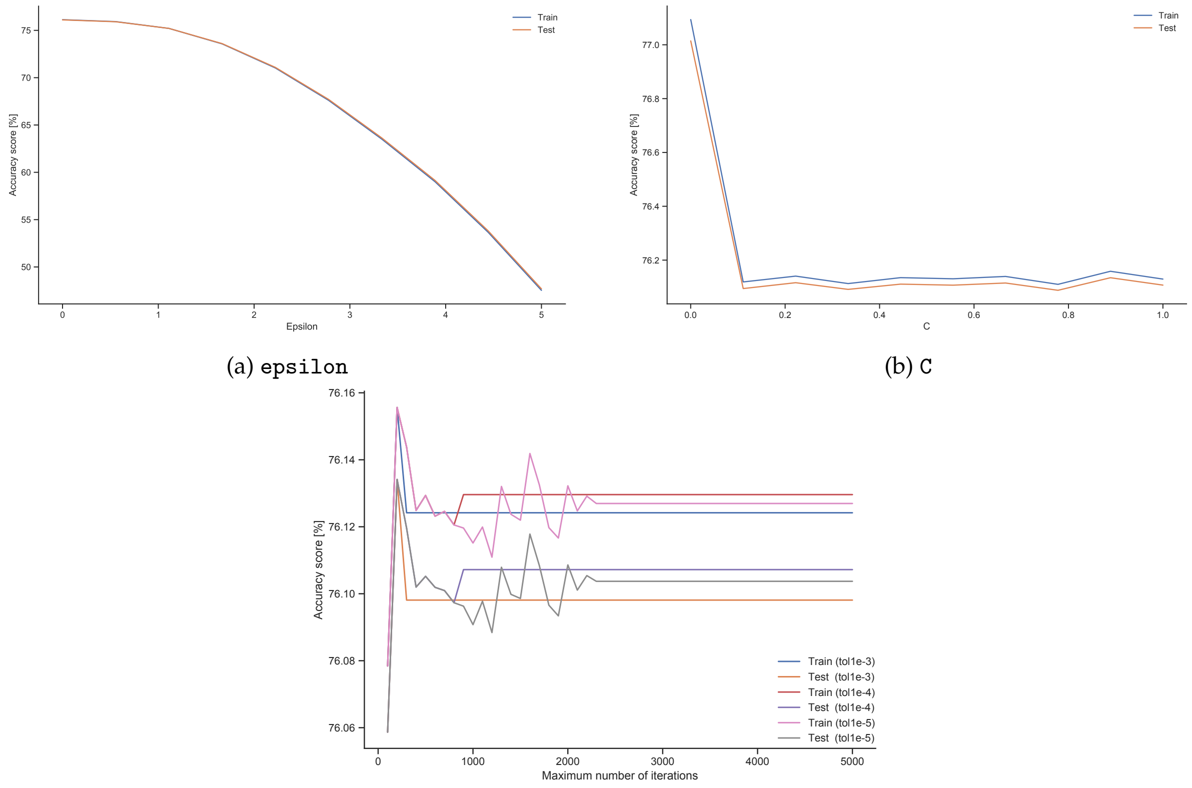
The *minimum samples for splitting* and *minimum samples in a leaf* do also influence how much the model overfits. Figure 4.9c and 4.9d both show that a low number of samples causes the model to overfit; the performance on the training set is a lot higher than on the testing set. As soon as the number of samples increases, the training and testing performance grow closer to each other.

For the *maximum number of features* and *bootstrap* hyperparameters, no graphs are plotted. The former allows setting its value to `auto`, `sqrt` and `log2`. When set to `auto`, the maximum number of features equals the total number of available features. For `sqrt` or `log2` it means that only the square root or the binary logarithm of the total number of features are used as the maximum number of features. Bootstrap is either set to `True` or `False` and if enabled, the samples are drawn with replacement as discussed in Section 3.3.1. Otherwise, all data is used for building the tree.

Hyperparameter	Description
`epsilon`	The epsilon value used in the ϵ-insensitive loss function
`C`	The regularisation parameter
`tol`	The tolerance for the stopping criteria of the optimisation
`max_iter`	The maximum number of iterations the algorithm should run for
`loss`	The type of loss function: ϵ-insensitive loss or squared ϵ-insensitive loss
`dual`	Whether to solve a dual or primal optimisation problem. If `n_samples` $>$ `n_feautres`, dual is preferred

Table 4.4: The hyperparameters for the SVR in the `scikit-learn` library.

For the hyperparameters in SVR (see Table 4.4), *epsilon* influences the width of the boundary of the loss function as described in Section 3.3.3. Figure 4.10a shows that an increase in the value leads to a significant decrease in model performance. An epsilon of zero can cause overfitting, but since its value is chosen based on the training set, it is better to choose a smaller value to allow the model to generalise better to new data.



(a) `epsilon`

(b) `C`

(c) `tol` and `max_iter`

Figure 4.10: Influence of the different SVR hyperparameters on the model performance for a training and testing set. These graphs are generated based on the rural and suburban data. 50% of the data is used for training, and the other 50% is used for testing.

The *regularisation parameter* (C) (see Figure 4.10b) influences the penalties applied to outliers outside of the ϵ-boundary. If its value is set too high, the regularisation decreases and overfitting occurs. Next, the *tolerance* and *maximum number of iterations* are evaluated. These two hyperparameters are combined into one graph because they both influence the convergence of the model. Figure 4.10c shows that for a lower tolerance value, it takes more iterations for the model to stabilise. A higher tolerance value means fewer optimisations of the model, resulting in slightly worse model performance.

Lastly, for the *loss* and *dual* hyperparameters no graphs are provided. The loss is either the ϵ-insensitive loss function or the squared ϵ-insensitive loss function, while dual is either `True` or `False`.

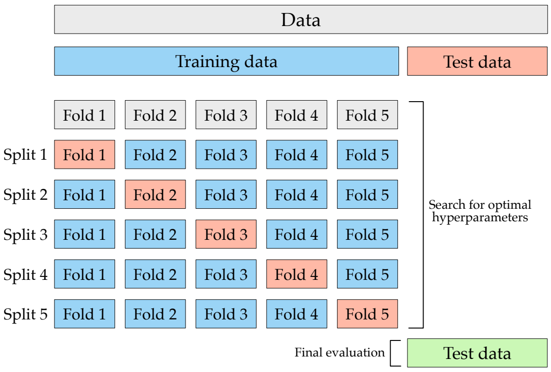


Figure 4.11: A 5-fold cross-validation process. Adapted from: scikit-learn.

Based on the findings in these graphs, search grids are created. I then apply a *randomised search* process. The process includes *k-fold cross validation*, with $k = 5$. The dataset is partitioned into five sets (i.e. folds) and for each fold the model is trained on the remaining four sets (see Figure 4.11) [Duan et al., 2003]. In each split, the model is validated based on the left-out data and an accuracy measure is computed. The performance of the 5-fold cross-validation process is the average of all individual evaluations. The approach makes sure that every data point is used for validation exactly once.

A consequence of the randomised search is that not all hyperparameter combinations are tried. A total of 75 hyperparameter combinations are sampled per fold. Ideally, the whole search space is tried, but given the many possible combinations, this is computationally expensive. At the end of the cross-validation process, the hyperparameter combination with the best score is selected. A separate test dataset can then be used to assess the model performance on unseen data. Table 4.5 shows the results of the hyperparameter tuning. For the split CBD and rural and suburban training data, only the nine geometric features are included. The combined training data includes the area morphology as an additional feature.

4.4 Feature Contributions to Height Predictions

The nine geometric features described in Section 3.4.1 should be analysed in terms of their usefulness for predicting building heights. Features showing a low contribution should be eliminated, which is also known as *feature selection* or *feature elimination*. It is preferred to select a subset of features that can still efficiently describe the input data and produce good prediction results [Guyon and Elisseeff, 2003]. This introduces several advantages [Zhu et al., 2007; Chandrashekar and Sahin, 2014]. First, redundant, irrelevant, or noisy data is removed. Second, the computational cost is reduced because during the training phase the algorithm needs to fit the model to fewer features. Third, the performance of the learning algorithm is improved, because the chances of overfitting based on the training set are reduced. Last, the complexity of the model is minimised, providing a better understanding of the data.

The analysis of the features is split into two parts. First, the relation between the features and the building height is analysed based on using two separate training datasets; one for the CBDs and one for

	Prediction model		
Hyperparameter	*CBD*	*Suburbs / Rural*	*Combined*
`n_estimators`	450	100	100
`max_depth`	14	`None`	`None`
`min_samples_split`	50	20	20
`min_samples_leaf`	15	5	5
`max_features`	`sqrt`	`sqrt`	`sqrt`
`bootstrap`	`False`	`True`	`True`

(a) RFR method

	Prediction model		
Hyperparameter	*CBD*	*Suburbs / Rural*	*Combined*
`epsilon`	1.0	0.0	1.0
`C`	1e-3	1e-4	1e-2
`tol`	1e-4	1e-5	1e-4
`max_iter`	1800	5000	200
`loss`	squared ϵ-insensitive	squared ϵ-insensitive	ϵ-insensitive
`dual`	`True`	`False`	`True`

(b) SVR method

Table 4.5: The optimal hyperparameters after tuning the RFR and SVR methods based on all nine geometric features. The combined model adds the area morphology as an extra feature.

the rural and suburban areas (see Section 4.4.1). Second, a combined training dataset is analysed where the area morphology is added as an extra feature in the training dataset (see Section 4.4.2).

Different approaches for feature selection exist and they can be divided in roughly three types: *filter*, *wrapper* and *embedded* methods [Bousquet et al., 2004; Chandrashekar and Sahin, 2014]. Filter methods are among the simpler techniques to use for feature selection problems. They only look at statistical measures and do not depend on any machine learning algorithm [Alamdari, 2006; Duch, 2006]. Often a measure is used to define how a single feature could influence the target variable, independently from any other features. The measure is applied to all features, and those with the highest values are selected as the best predictors for the target variable. The method assumes that better features are assigned higher values.

Wrapper methods are based on machine learning models and they do often make use of cross-validation to evaluate subsets of features based on feedback from the model (e.g. accuracy) [Duch, 2006]. The training stage makes this approach more computationally expensive than filter methods. Finding the optimal combination of features can be difficult, and sophisticated wrapper methods are needed to do so [Bousquet et al., 2004]. A drawback of wrapper methods is that they have a higher chance of overfitting because the subset of features is selected based on specific training data and a machine learning model.

Lastly, embedded methods provide a combination of the filter and wrapper methods, as the feature selection is embedded in the training phase of the machine learning algorithm [Duch, 2006; Chandrashekar and Sahin, 2014]. The importance of the features is derived from the trained model, and features with low importance are removed. Tree-based models, such as RFs, can be used for such tasks (see Section 3.3.1).

In this thesis project, I will mainly focus on filter methods, because they are less computationally expensive and have a lower chance of overfitting based on the training data. Also, the process is independent of any machine learning algorithm, which leads to similar subsets of features for all methods instead of algorithm-specific subsets that perform best for the given method. For comparison reasons, the results of the feature importance of a RF are compared to the outcomes of the filter methods to check for similarities and deviations.

The correlation between the independent features and the building height is an example of a filtering method. For capturing linear correlations between features, two different approaches can be used. First, a correlation matrix can be created based on Pearson's correlation coefficient [Guyon and Elisseeff, 2003; Chandrashekar and Sahin, 2014]. Its values range from -1 to 1, where -1 indicates a strong negative correlation, 0 no correlation, and 1 a high positive correlation between the features. Second, an F-test can be used to compute the F-scores of the different features, where a higher score indicates a higher linear relationship between the two entities. The F-test looks at how significantly the model improves when new variables are added, and it uses the residual sum of squares as an error measure to do so [Lomax and Hans-Vaughn, 2013]. Then, these two measures can be used in *univariate selection*, where single features get ranked and the k-best features are selected to perform the predictions with [Jović et al., 2015].

A drawback of both the Pearson's correlation coefficient and the F-test is that they are only sensitive to linear relationships [Guyon and Elisseeff, 2003; Lomax and Hans-Vaughn, 2013]. Non-linear relations cause the Pearson's correlation coefficient to be close to zero, even if the two features have a one-to-one correspondence. An alternative would be to use mutual information (MI), which measures the dependence of one variable to another. In contrast to Pearson's correlation coefficient and the F-test, it can capture non-linear relations between the variables [Chandrashekar and Sahin, 2014]. An MI of 0 means variables are independent, and a score of 1 means they are fully dependent. The interpretation of a non-zero value means that a variable X can provide information about a variable Y, and thus they are dependent. However, since two of the machine learning methods (MLR and SVR) in this thesis are based on linear relationships, it is not sensible to search for non-linear relationships in the data because the methods will not be able to capture such relations.

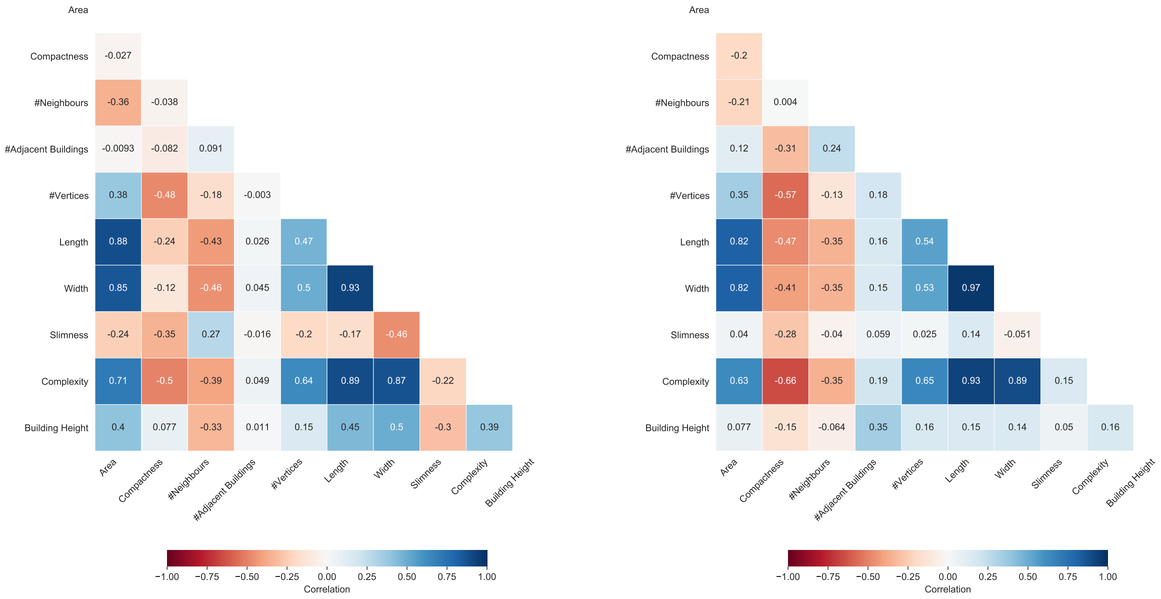
4.4.1 Separate Training Data

The correlation matrices in Figure 4.12 show Pearson's correlation coefficient applied to the CBD and the suburban and rural areas. The bottom row in the matrix provides the correlation between the different features and the building height. The correlations clearly differ for the two area types. For the CBDs (see Figure 4.12a), the correlations show that multiple features have a significant positive relationship to the building height. These include the building width, length, area, and shape complexity. The number of neighbours and building slimness show a mentionable negative correlation to the building height. For the suburban and rural data, fewer extremes are present in the correlation outcomes (see Figure 4.12b). Only the number of adjacent buildings seems to have a relatively high impact on the building height. The number of vertices in the building footprint, shape complexity, and the building length and width show a low positive correlation to the building height. In contrast, the shape compactness expresses a low negative correlation.

Comparing these correlation values to the outcomes of the F-test shown in Figure 4.13 makes clear that both methods show similar trends for both area morphologies. The bar chart makes the difference between the CBDs and the suburban and rural areas even clearer. Selecting the five best features from these two filter methods provides the same outcome: for the CBDs the building width, length, area, shape complexity, and number of neighbours are selected, while for the suburban and rural data the number of adjacent buildings, number of vertices in the footprint, shape complexity, length and the compactness are most important.

A problem with selecting the features in such a way is that the collinearity between features is not removed. Excluding these closely related features is especially important in linear regression problems, as one of its assumptions is that the independent variables are uncorrelated with each other [Hill and Adkins, 2003; James et al., 2013]. If left unchanged, it becomes a challenge to determine how each of the features is separately associated with the target variable. Also, the chances of overfitting are higher because the data contains more noise. Similar problems arise for support vector regressors with linear kernels because of their dependence on linear functions [Vapnik, 1995].

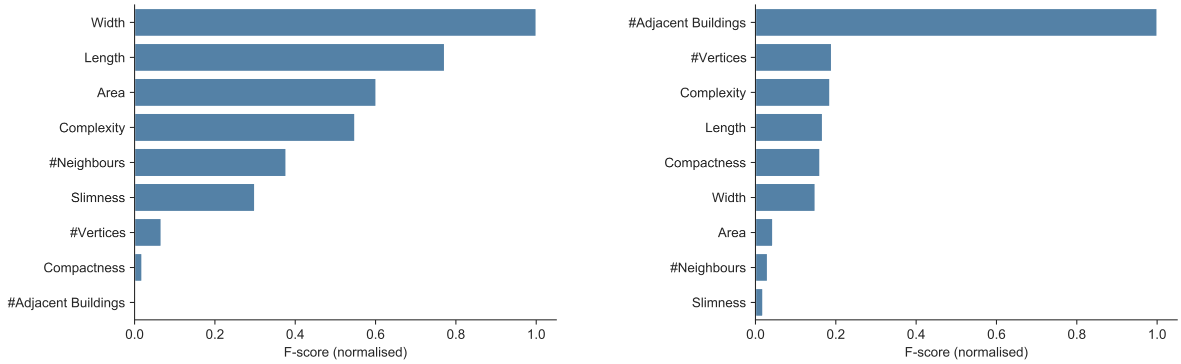
For RFs, no assumptions about the distribution of the data are made because they are of a non-parametric nature [Breiman, 2001]. However, correlated features do influence the permutation importance measure of the different features [Toloşi and Lengauer, 2011; Gregorutti et al., 2017]. Mainly, the most discriminating correlated features might not have higher importance values than the less discriminating features. The group size of the correlated features impacts these results. A larger group size means a larger shared responsibility of the features in the model, resulting in lower weights. So, features that are highly correlated with the target variable may appear insignificant in large groups.



(a) CBD data

(b) Suburban/ Rural data

Figure 4.12: Correlation matrices for the different types of areas in the building footprint training data.



(a) CBD data

(b) Suburban/ Rural data

Figure 4.13: The F-score for each feature in the two area morphologies based on univariate linear regression tests. A higher score means the feature has a higher degree of linear dependency with the building height.

When three or more features express collinearity among each other, we are dealing with multicollinearity. Detecting multicollinearity based on the values in a correlation matrix is difficult. Computing the Variance Inflation Factor (VIF) is a more suitable method to quantify and asses the multicollinearity of a set of variables [James et al., 2013]. The index describes the increase in the variance of an estimated regression coefficient caused by collinearity. VIF has a smallest possible value of 1, which is when there is a complete absence of collinearity. Values exceeding 5 or 10 indicate high collinearity between variables, which can be problematic.

Table 4.6 shows the VIF scores for the nine features in the CBDs and the suburban and rural areas. The building length, width and shape complexity stand out from the rest with some very high VIF scores. For the CBDs, the building area is just above the limit of 5, while for the suburban and rural areas it is just under this limit. All other features show acceptable VIF scores between 1-5.

The VIF scores for the features, after selecting the five best features based on their correlation to the building height, are high. This is especially true for the CBD data, all but one score are above the limit of 5. The suburban and rural data shows a bit less collinearity among the features, but the shape complexity and building length are still problematic. The analysis makes clear that it is difficult to select a subset of features that is highly correlated to the target variable, and at the same time does not express high collinearity among the features in this subset.

	CBD		**Suburban / Rural**	
Feature	*All*	*5 best*	*All*	*5 best*
Area	6.32	6.11	4.66	-
Compactness	3.61	-	2.91	2.47
Complexity	16.50	6.78	18.91	13.61
#Neighbours	1.35	1.28	1.42	-
#Adjacent buildings	1.04	-	1.21	1.11
Length	26.49	14.63	48.27	9.23
Width	24.02	8.80	40.17	-
Slimness	3.73	-	2.56	-
#Vertices	2.02	-	1.96	1.89

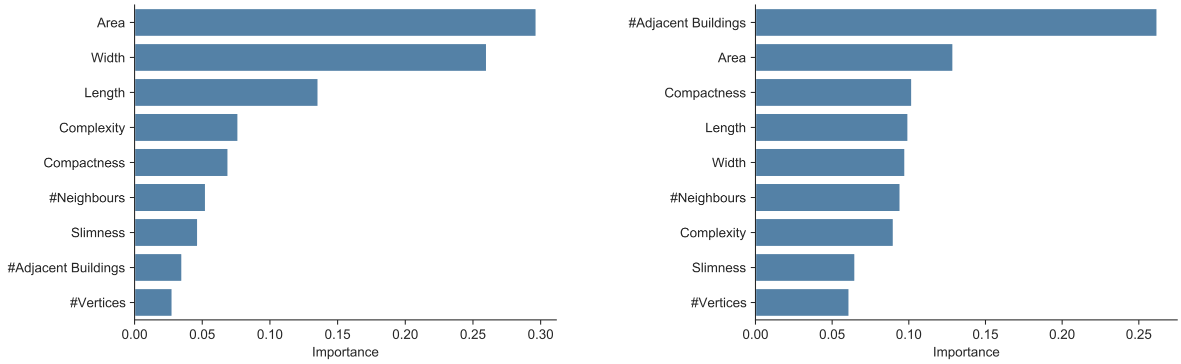
Table 4.6: VIF scores for the features in the two area morphologies.

When deriving the feature importance from a RF, two types of feature importance can be used. First, the *impurity-based* feature importance is based on how well the trees split the data [Nembrini et al., 2018]. The splitting rules of RFs try to maximise the impurity reduction that is introduced by a split. A split causing a large impurity reduction is deemed important. Consequently, the features used in this split are also considered important. Bias is present towards features with many possible split points (e.g. numerical data). The ability of a feature to be useful in future predictions cannot be derived as it only reflects its importance in the training set.

Second, the *permutation importance* is the feature importance as established by Breiman [2001], where the importance of a feature is defined as the decrease in a model score when a random shuffle of the single feature value is performed. The relation between the feature and the target variable is broken in this process. The feature gets randomly shuffled n-times and then a sample of the feature importances is returned. The bias present in the impurity-based feature importance is not present in the permutation importance.

Figure 4.14 shows the impurity-based feature importances for the nine geometric features for the CBDs and the suburban and rural areas. The permutation importance for the same features, applied to a training and testing set (80/20 division of the full data), is shown in Figure 4.15 and 4.16. Each feature is permuted 25 times and the results are shown in a boxplot, where the median value is the vertical orange bar in the box and the circles indicate outliers. If features are highly important on the training set, but not on the test set, this might indicate that they can make the model overfit. The order of the features makes clear that the same ranking is present in both the training and test sets. Comparing the impurity-based importance with the permutation importance of the training set, we do see that the ranking of the features is different for both the CBDs and the suburban and rural areas.

Comparing the top five features from the correlation-based filter methods to the permutation importances of the RF, quite some differences are present, especially for the suburban and rural regions. For the CBDs, the width, length, area and number of neighbours are present in both subsets. The filter method prefers the shape complexity, while the permutation importance prefers compactness. Both provide a measure for the building footprint boundary, and the correlation matrix (see Figure 4.12a) shows they express a high negative correlation. The VIF score of the compactness is much lower than that of the shape complexity, making it a possibly better predictor because of the lower collinearity with other features.



(a) CBDs

(b) Suburban / Rural areas

Figure 4.14: Impurity-based feature importance for the two area morphologies derived from a RF.

The suburban and rural regions show significant differences between the filter and permutation importance methods. The number of adjacent buildings is the most important feature for both, but apart from this only the shape complexity is present in both subsets. However, the building length and width are so highly correlated to each other (see Figure 4.12b) that it is possible to use either one of them, but preferably not both at the same time.

Table 4.7 provides an overview of the top five features per method including their VIF score to highlight the multicollinearity among the features. It is clear that especially the length, width and shape complexity introduce multicollinearity among the features.

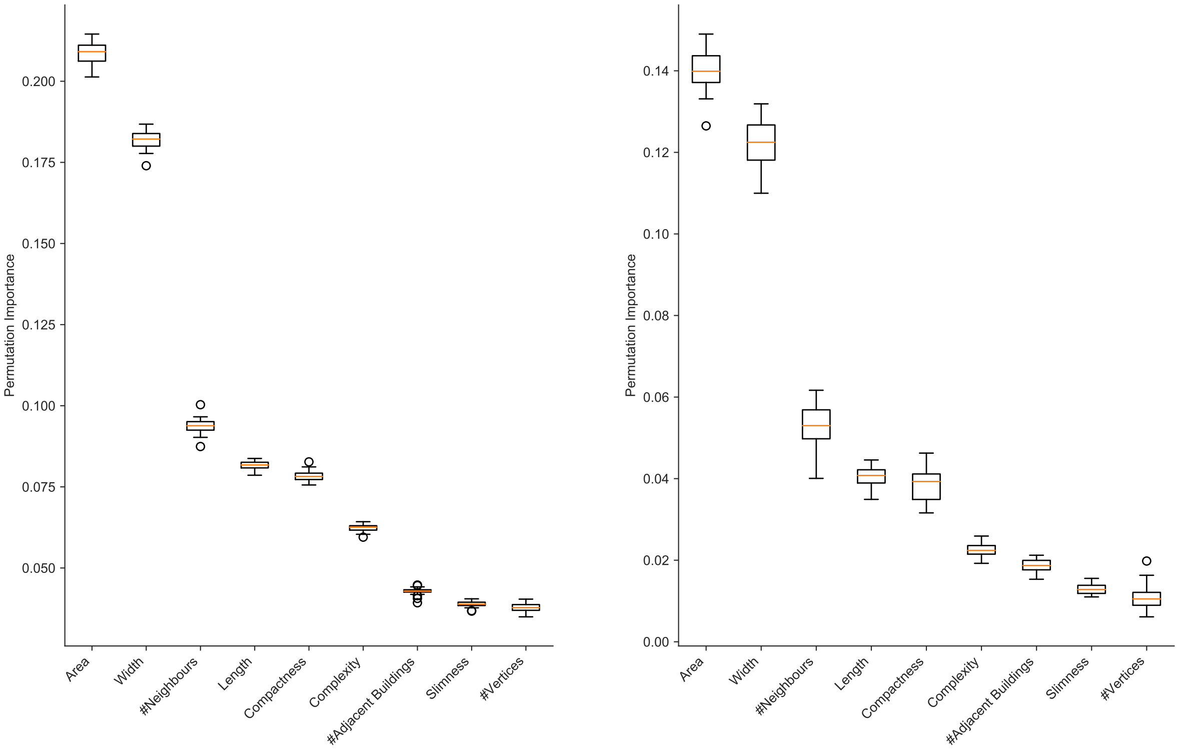
	CBD				**Suburban / Rural**			
Rank	*Correlation*	*VIF*	*Permutation*	*VIF*	*Correlation*	*VIF*	*Permutation*	*VIF*
1.	Width	8.80	Area	5.65	#Adjacent	1.11	#Adjacent	1.18
2.	Length	14.63	Width	7.88	#Vertices	1.89	Area	3.61
3.	Area	6.11	#Neighbours	1.31	Complexity	13.61	#Neighbours	1.31
4.	Complexity	6.78	Length	11.98	Length	9.23	Complexity	6.10
5.	#Neighbours	1.28	Compactness	1.38	Compactness	2.47	Width	10.88

Table 4.7: The top five features based on the filter method and permutation importance. For each feature the VIF score is provided to highlight multicollinearity between the features.

The analysis shows that selecting a subset of features suitable for predicting building heights is a challenge. However, it looks like that certain features can provide an indication of the building height, while also avoiding collinearity with other features. Section 5.6 includes an overview of several prediction models trained on different feature subsets, including the accuracy of the predictions.

In addition to these numerical analyses, we can visually inspect the distribution of the nine geometric features given the building heights. A method for inspecting such relations are violin plots. It is similar to a box plot, but it adds a rotated kernel density plot showing the probability density of the data at different values. Figure 4.17 and 4.18 show the violin plots for the CBDs and the suburban and rural data respectively. The height on the y-axis is cut off at 300m for the CBDs and at 150m for the suburban and rural data to make the shapes of the distributions clearer (see Appendix C for the plots including the full height range). The continuous values of the nine geometric features are divided into five categories of equal size. The white dot inside each of the violins represents the median height for that category. If the shape of the plots is similar for each category within a feature, it becomes more difficult for a machine to distinguish between those categories.

For the CBDs in Figure 4.17, the distribution of the building area differs for the five categories making it a possibly good predictor. For the compactness, the last three groups display similarities, introducing possible difficulties. The number of neighbours shows even more similarities between the categories, while the number of adjacent buildings is quite spread out. However, the median height per category is very similar. For the number of vertices, a high density of values is present at a similar building height, making it hard to distinguish between the groups. The length and width both show similar trends in the different categories: they are quite spread out and seem to differ, also in the median height. The building slimness, however, shows high peaks around the same height for four of the categories. Lastly, the shape complexity does show varying distributions, but again there is a high density around the same building height.



(a) Training set

(b) Testing set

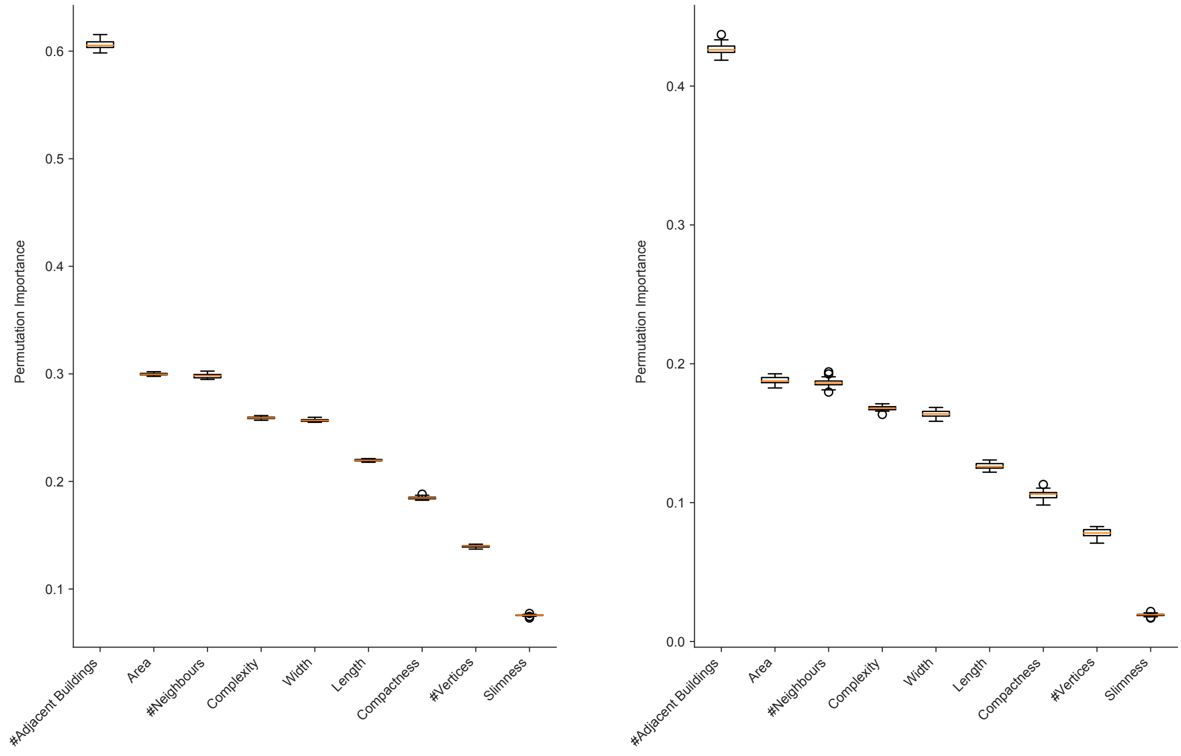
Figure 4.15: Permutation importance for the CBDs derived from a RF after 25 shuffles. 80% of the data is used for training, and 20% is used for testing.

Relating these findings back to the top five features of the approaches shown in Table 4.7, provides interesting insights. The area, width, length, shape complexity, number of neighbours, and compactness are regarded as the most important features when combining the top five for the correlation and permutation methods. Most of these features show varying distributions between the categories of the violin plots, supporting the findings of the filter methods.

A similar analysis can be made for the suburban and rural data in Figure 4.18. At a glance, it is already clear that there are quite a few features that show comparable trends among their range of values within the different features. This holds especially for the building area, the number of neighbours, length, width and shape complexity. The compactness shows for the first two categories differing distributions, but the latter three groups are again quite similar. For the number of adjacent buildings, the distributions are very different for the five categories, making it a possible good predictor for the building height because clear distinctions can be made between values. The number of vertices does show different density distribution shapes for the categories, but the median height is again very similar.

When comparing these findings to the top five features of the approaches in Table 4.7, we see that the correlation and permutation approach do not agree as much as for the CBD data. For both methods, the number of adjacent buildings is the most important feature, which is also the feature that has the most distinct distributions between the categories in the violin plots. For the other features, no such clear relations are present.

These density distribution plots make clear that it can be difficult for a machine to detect clear patterns within the values of the different features, which can lead to lower performance of the machine learning algorithms. However, deriving features from only the 2D geometries of the building footprints imposes limitations because only a limited number of characteristics can be derived.



(a) Training set

(b) Testing set

Figure 4.16: Permutation importance for the suburban and rural areas derived from a RF after 25 shuffles. 80% of the data is used for training, and 20% is used for testing.

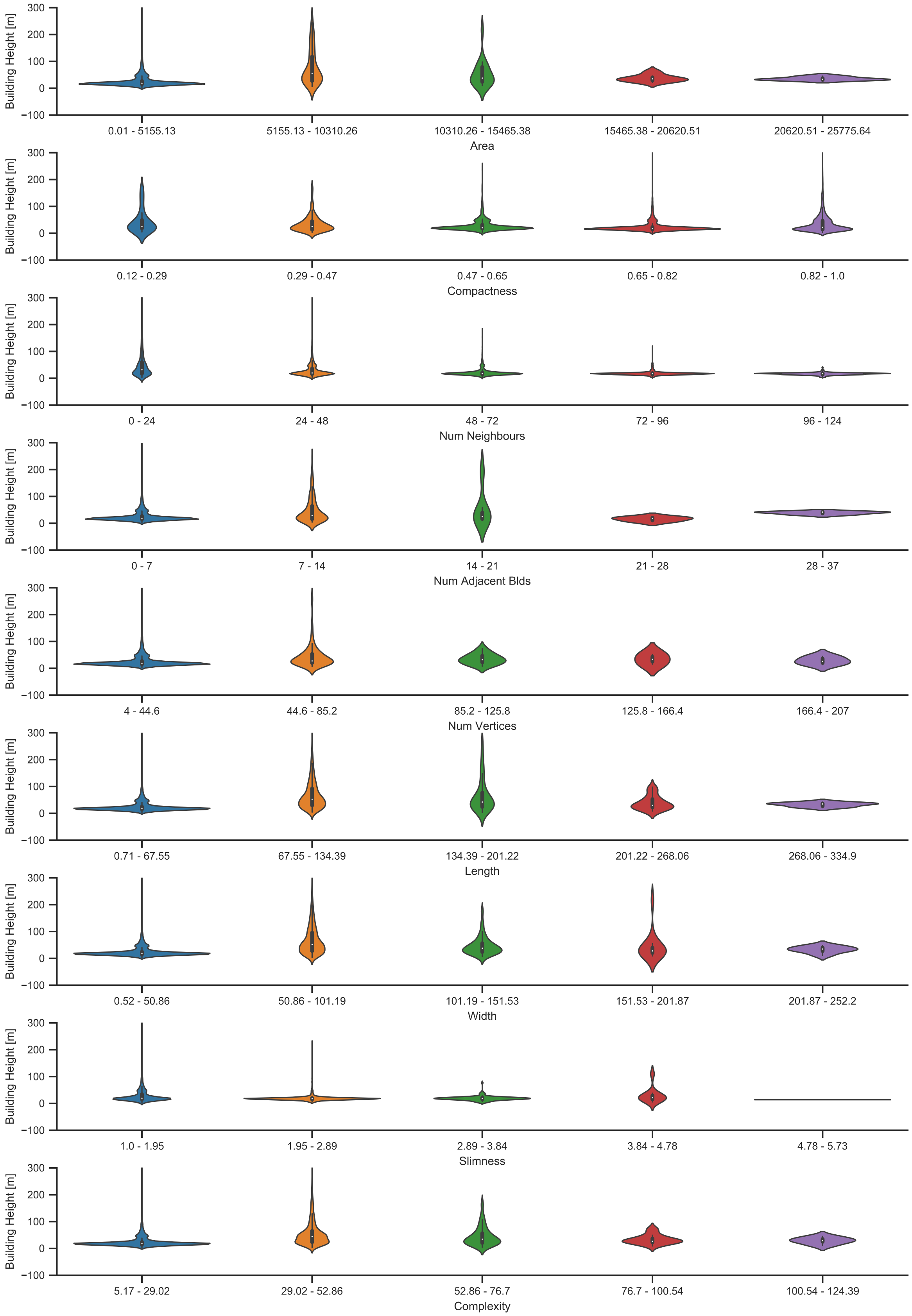


Figure 4.17: Violin plot showing the probability density of the building height for each of the features in the CBD setting. The building height is cut off at 300m to make the density distribution clearer.

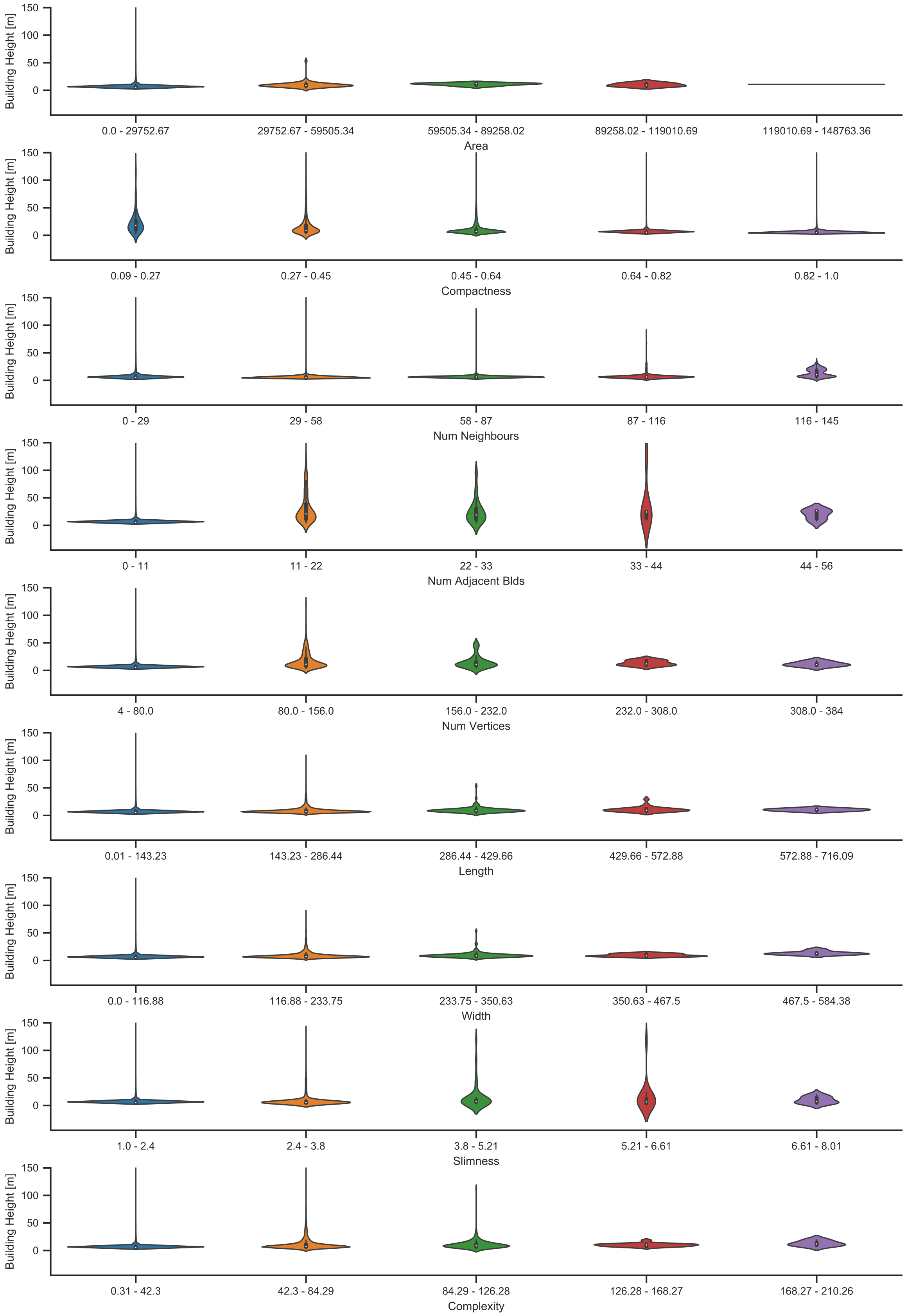
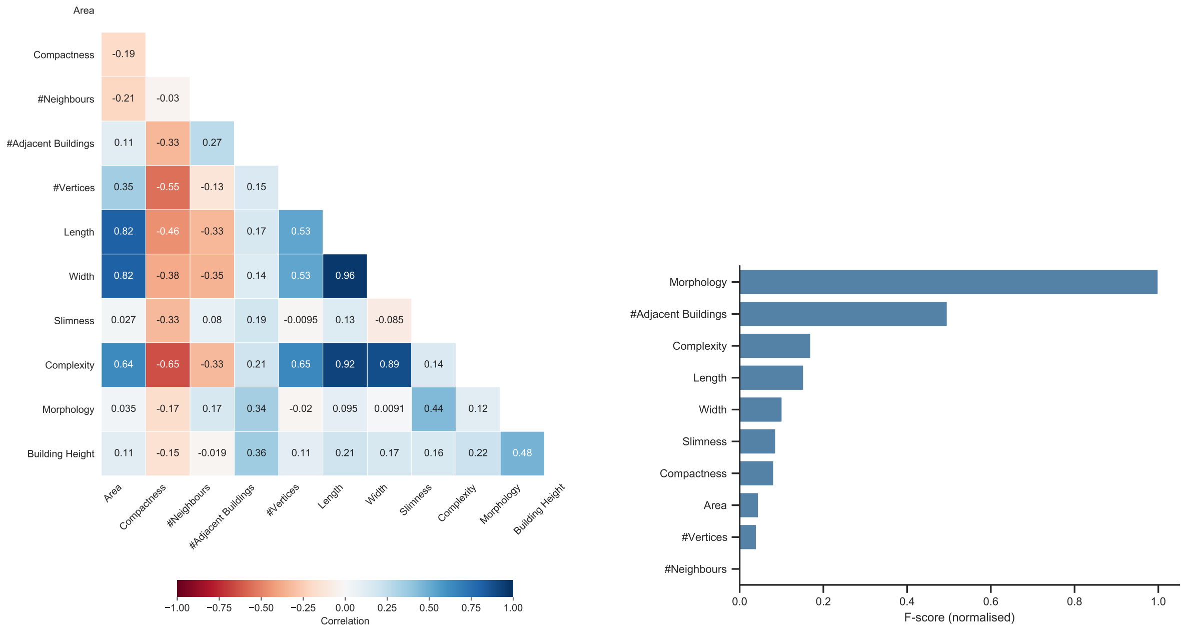


Figure 4.18: Violin plot showing the probability density of the building height for each of the features in the suburbs/ rural setting. The building height is cut off at 150m to make the density distribution clearer.

4.4.2 Combined Training Data

Next, a similar analysis is performed for the training dataset that combines the training data from both area morphologies, and which uses the morphology as an extra feature in the training phase. Figure 4.19a shows Pearson's correlation coefficient for the different features. The trends are quite similar to those in the rural and suburban areas shown in Figure 4.12b. For most features, the linear relationship between the features and the building height did slightly increase. But, as was the case with the rural and suburban areas, the correlation values are not significantly high. It is clear, however, that the area morphology feature shows a strong linear relationship to the building height with a correlation coefficient of 0.48. This makes it the most important feature in terms of Pearson's correlation values. The scores of the F-test in Figure 4.19b show similar trends to the values in the correlation matrix. From these two sources of information, the five best features are the area morphology, number of adjacent buildings, shape complexity, length and width.



(a) Correlation matrix

(b) F-scores

Figure 4.19: Results of the two filter-based approaches for the combined training dataset. Both methods express the linear relationship between the features and the building height.

As stated in Section 4.4.1, the collinearity between the features is not removed in this process. The Variance Inflation Factor (VIF) score the features is shown in Table 4.8. The table also shows the scores if only the top five features are kept. In the top five, all scores, except those of the number of adjacent buildings and the area morphology, are exceeding the limit of 5. Similar trends as for the CBDs and the rural and suburban areas are observed. The area morphology, however, was not present as a feature in the split training data. The results in Table 4.8 make clear that it has a very low VIF score.

Next, we look at the two types of feature importance that are derived from a RF: the impurity-based and permutation importance. The impurity-based feature importance is shown in Figure 4.20, while the permutation importance is shown in Figure 4.21. The latter is split up in a training and testing set (80/20 division of the full data), and each feature is permuted 25 times to generate the boxplots. The median value is indicated by the vertical orange bard in the box, and the circles indicate outliers. Both the training- and testing set show the same ranking of the features, which means a lower chance that the model is overfitting. Comparing the ranking to the impurity-based importance ranking in Figure 4.20, there are some small differences present. In the top five features, the width has been replaced by the number of neighbours in the permutation importance. For the bottom five features, the number of vertices and footprints slimness swapped places. All in all, the differences are less extreme than for the split training data shown in Section 4.4.1.

	VIF	
Feature	*All*	*5 best*
Area	4.63	-
Compactness	3.01	-
Complexity	18.31	6.94
#Neighbours	1.45	-
#Adjacent buildings	1.34	1.17
Length	44.50	19.95
Width	36.55	14.42
Slimness	3.14	-
#Vertices	1.95	-
Morphology	1.40	1.23

Table 4.8: VIF scores for the features in the combined area morphology data.

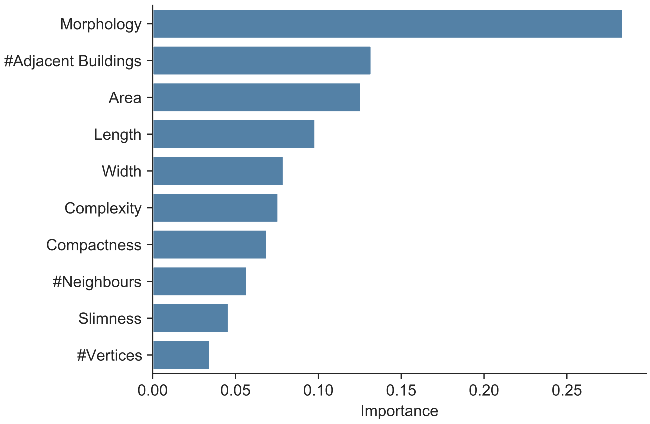
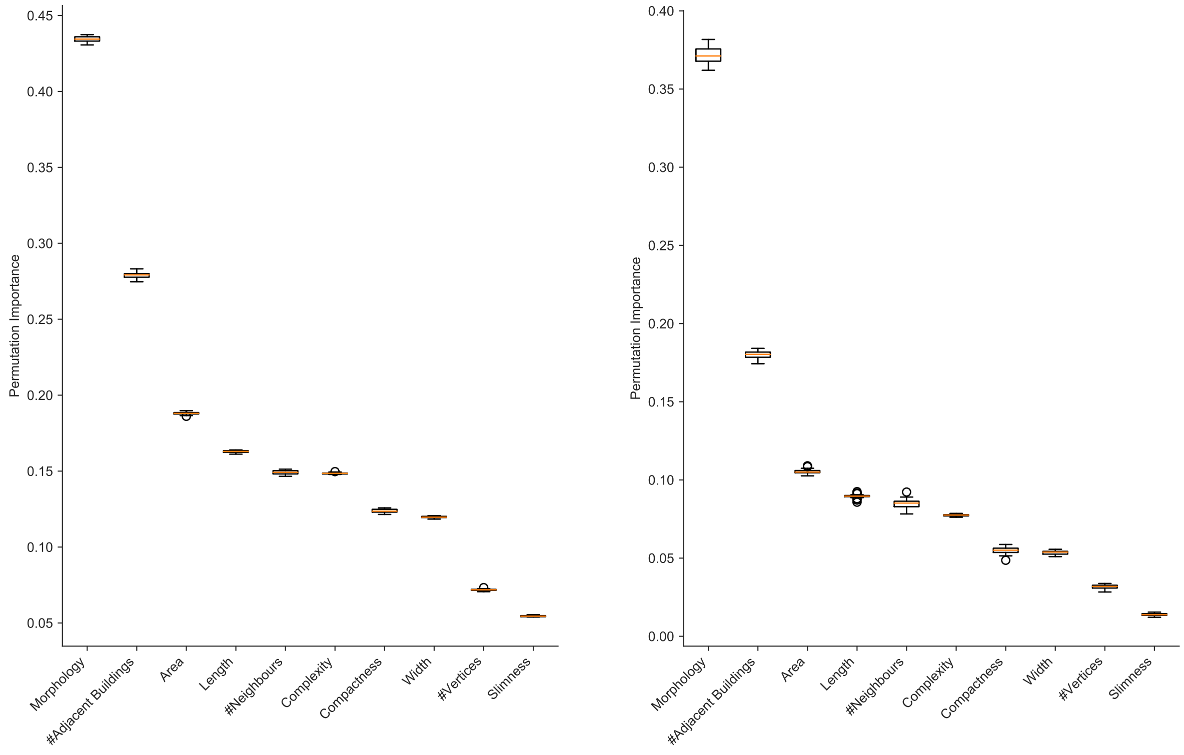


Figure 4.20: Impurity-based feature importance for the combined area morphologies derived from a RF.

The comparison between the top five features from the correlation-based filter methods and the permutation importance of the RF highlights some small differences (see Table 4.9). The morphology, number of adjacent buildings and length are present in both methods, but the filter methods prefer the shape complexity and width while the permutation importance prefers the area and number of neighbours. What stands out are the VIF scores for the top five features based on the permutation importance: all these values are below the limit of 5, indicating there is no significant multicollinearity present among the features. The top five features of the correlation importance do express quite a high multicollinearity, especially among the shape complexity, length and width. Based on these findings, subsets of features will be used to train different prediction models to investigate their influence on the accuracy of the building height predictions (see Section 5.6).

Lastly, a visual inspection of the distribution of the features given the building height is performed using violin plots. Figure 4.22 shows the violin plots for the combined training data, where the height on the y-axis is cut off at 200m to make the shapes of the distributions clearer (see Appendix C for the plots including the full height range). The continuous values of the nine geometric features are again divided into five categories of equal size, while the morphology is already categorical. The white dot inside each of the violins represents the median height for that category.

What stands out from the violin plots is that several features show comparable trends, making it possibly more difficult to differentiate between these features. The density plots of the five categories of the area, number of neighbours, width, length and shape complexity are all relatively flat with median values at roughly the same height. The compactness shows different densities for the first three groups, but the last two are again very similar. For the number of adjacent buildings, there is a lot more variety between the five categories and their median heights clearly differ. This can make it a potentially good feature for the building height predictions. For the number of vertices, the density plots slightly differ in shape, and the first two groups express higher building heights. However, the median values are again quite similar. A similar observation is present for the building slimness. Lastly, for the area morphology the CBDs do clearly contain more buildings with higher heights.



(a) Training set

(b) Testing set

Figure 4.21: Permutation importance for the combined data derived from a RF after 25 shuffles. 80% of the data is used for training, and 20% is used for testing.

Rank	Correlation	VIF	Permutation	VIF
1.	Morphology	1.23	Morphology	1.15
2.	#Adjacent	2.27	#Adjacent	1.28
3.	Complexity	6.94	Area	3.10
4.	Length	19.95	Length	3.50
5.	Width	14.42	#Neighbours	1.33

Table 4.9: The top five features based on the filter method and permutation importance for the combined area morphology data. For each feature the VIF score is provided to highlight multicollinearity between the features.

Relating these findings back to the top five features of the approaches shown in Table 4.9, we see that the morphology and the number of adjacent buildings are most important in both methods, and the violin plots show that the different groups should be separable. However, for the shape complexity, length, width, and the number of neighbours the data distributions are generally quite similar for all categories. This can indicate that the machine learning algorithms might struggle to find clear relations to the building height for these features.

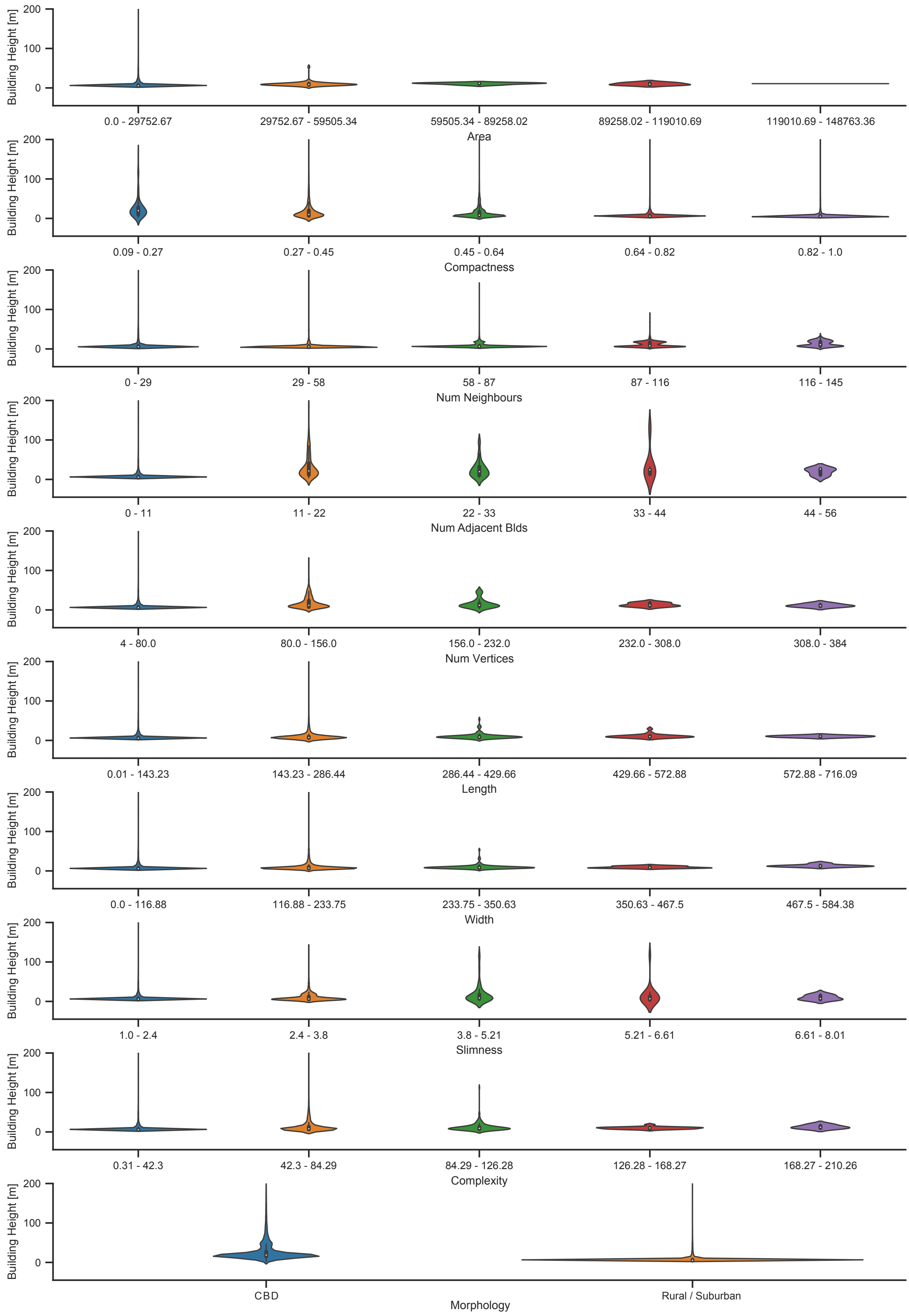
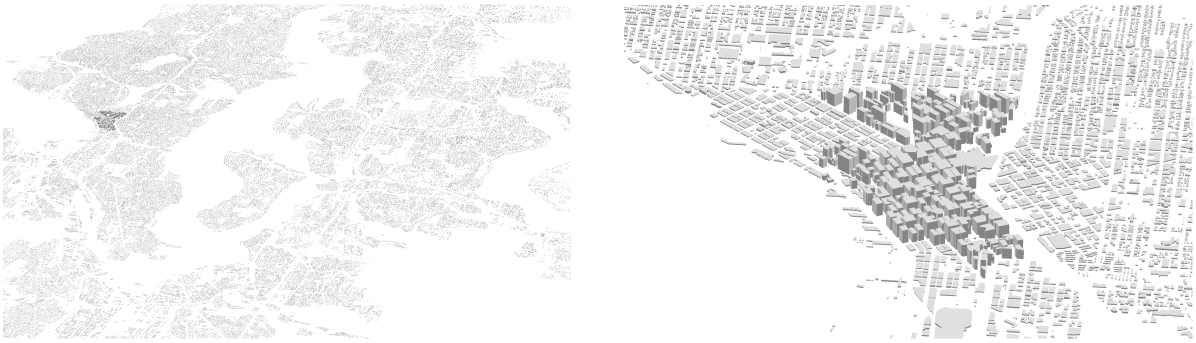


Figure 4.22: Violin plot showing the probability density of the building height for each of the features in the combined area morphology data. The building height is cut off at 200mm to make the density distribution clearer.

5 Results & Analysis

In this chapter, the results of the implementation of the methodology are presented and discussed. An example of the results is shown in Figure 5.1, which provides a 3D city model of Seattle based on the height predictions of the RFR machine learning method. The time it takes to train the models is reviewed first, together with the duration of the building height predictions for the conterminous USA (Section 5.1). Next, an analysis of the accuracy of the height predictions is performed for three test areas, where a distinction is made between two types of prediction models (Section 5.2). The first uses two separate prediction models for the different area morphologies, while the second uses only one model but it adds the morphology as an extra feature for each building footprint. Excluding the area morphology from the training and predicting stages is analysed in Section 5.3. The results of the three machine learning methods are then compared to the Open City Model (OCM) (Section 5.4). In Section 5.5 it is investigated whether adding additional features to the prediction models improves the prediction results. This is tested on the test area of Denver, Colorado. Next, different subsets of features are analysed in terms of their performance (Section 5.6). Lastly, Section 5.7 provides a review of the influence of different height percentiles on the test results.


(a)

(b)

Figure 5.1: 3D city model of the results for Seattle, Washington, using the height predictions from the RFR method. The prediction model is trained on only the suburban and rural training data. (a) provides a general overview, while (b) shows a close-up of the CBD.

5.1 Methodology Runtime

The runtime of the methodology is reviewed based on two aspects: 1) the time it takes to train the prediction models, and 2) the time it takes to predict the building heights for all building footprints in the conterminous USA. The two area morphologies are considered during both phases. Table 5.1 shows the training time for the different models, which is computed as the average over ten test runs. The CBD and the suburban and rural models include all nine geometric features, while the combined model adds the area morphology as an extra feature in the training process. For the RFR and MLR methods, the processes are run in parallel on 20 processors (see Section 4.2 for the server details). The results show that the MLR and SVR methods are trained the fastest. For the separate prediction models, the training time stays under a second for both area morphologies. In the combined model, the MLR method is still able to be fully trained in under a second, while the SVR method takes 2.23s. The time it takes to train the RFR method is significantly longer. However, 13.96s for the rural and suburban model, 2.62s for the CBDs, and 17.03s for the combined model are still not extremely high.

The time it takes to predict the building height for the roughly 125 million buildings in the conterminous USA is shown in Table 5.2. With the split model, the MLR and the SVR methods take around ten seconds to predict the building heights. When the area morphology is added as an extra feature in the combined model, the time lies more around 25 seconds. The RFR method is again the slowest of the three approaches. For the split model, the prediction time lies around 3.5 minutes, and for the combined model it lies around 6 minutes. Considering the number of building footprints in the USA dataset, these times are still very reasonable. The prediction times that are shown here do not include reading the data from the database into `Python`. This process takes around an hour, which is the average time over the six prediction test runs.

	Training time* [s]		
Regressor	*Suburban / Rural model*	*CBD model*	*Combined model*
RFR	13.96	2.62	17.03
MLR	0.12	0.01	0.50
SVR	0.70	0.04	2.23

* Average of 10 runs

Table 5.1: The training times for the three machine learning methods based on the different prediction models. Training is performed on the data described in Table 4.1. All nine geometric features are included, and for the combined model the area morphology is added as an extra feature. Only buildings $\geq$ 3m are considered. The RFR and MLR methods are run in parallel on 20 processors.

	Predicting time [mm:ss]	
Regressor	*Split model*	*Combined model*
RFR	03:38.03	05:56.08
MLR	00:09.73	00:23.13
SVR	00:10.55	00:26.91

Table 5.2: The prediction times for the three machine learning methods based on different prediction models. The predictions are performed for all building footprints in the conterminous USA. All nine geometric features are included, and for the combined model the area morphology is added as an extra feature. The RFR and MLR methods are run in parallel on 20 processors.

Given the results of the three methods, expanding the number of building footprints in the training process — to increase the variety in buildings — should not pose any problems. It could make the prediction models generalise better, improving the prediction accuracy.

These tests did not consider the process of storing the prediction results back into the database again. This process is often slower than reading the data from the database — especially for large amounts of data — because we have to write data to disk instead of only reading it.

5.2 Model Accuracy

The next step is the analysis of the height predictions for the three test areas. The accuracy results of these tests are shown in Table 5.3, where a division is made between the different types of regressors. For each area, two types of prediction models are tested. The first model is trained with data from only one area morphology, while the second model is trained on all training data and the morphology is used as an extra feature in the training and prediction process. All nine geometric features are included during these tests. The results show that there is not one machine learning method that performs best in all test scenarios. The same observation holds for the different types of prediction models. In the following subsections, the focus is put on the individual test areas and their results. For each area, the results of the machine learning methods are compared to the reference model that is created using LiDAR data.

Regressor		Seattle		Portland		Astoria	
		CBD	*Combined*	*Suburban / Rural*	*Combined*	*Suburban / Rural*	*Combined*
RFR	*MAE [m]*	40.54	39.74	1.42	1.42	2.29	2.29
	MAPE [%]	224.93	216.87	24.77	24.92	28.90	28.91
	RMSE [m]	48.58	47.91	2.36	2.36	2.99	2.98
	RMSPE [%]	361.21	351.15	32.64	32.76	36.00	35.95
MLR	*MAE [m]*	37.09	32.84	1.67	1.77	2.28	2.30
	MAPE [%]	218.27	117.57	27.27	29.30	29.30	29.91
	RMSE [m]	44.73	49.66	2.61	2.68	2.93	2.94
	RMSPE [%]	341.09	186.50	32.59	36.72	35.04	36.32
SVR	*MAE [m]*	36.83	34.88	1.65	1.41	2.27	2.51
	MAPE [%]	216.12	78.68	26.79	22.64	29.08	30.13
	RMSE [m]	44.44	55.25	2.58	2.39	2.92	3.21
	RMSPE [%]	337.03	107.51	31.91	26.64	34.58	34.21

Table 5.3: Accuracy results for the three different test areas for the different types of regressors. The prediction models are based on only CBD data, suburban / rural data, or on a combination of both. The first two include all nine geometric features, the latter also includes the are morphology as an extra feature.

5.2.1 Seattle

The first area is the detected CBD of Seattle, Washington. Table 5.3 shows that for the model trained on the CBD data, the SVR method achieves the lowest MAE of 36.83m. This is still a significant error, emphasising that the model highly under- and overestimates the building heights. The MAPE of 216.12% and the RMSPE of 337.03% support this observation. For the combined prediction model, the MLR method works best and it achieves the highest accuracy improvement compared to the model trained on only the CBD data. It has an MAE of 32.84m, which is still very high. The MAPE and RMSPE did drop significantly for both the MLR and SVR methods, but the combined model does not achieve these kind of improvements for the RFR approach. For Seattle, the prediction model that is trained on all data and uses the area morphology as an extra feature works best.

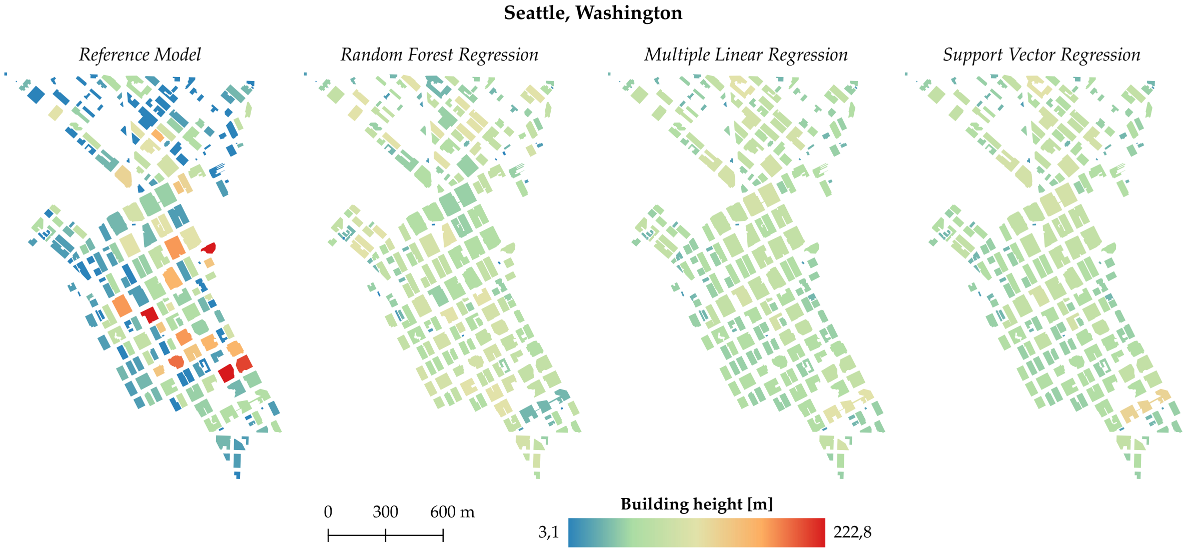


Figure 5.2: Maps showing a comparison between building heights in the reference model and the building height predictions of the three machine learning models for the CBD of Seattle, Washington. The prediction models are trained on only the CBD data.

The maps in Figure 5.2 provide a comparison between the three machine learning methods and the reference model. The predictions are performed with the model that is trained on only the CBD data. Buildings in blue express a lower height, while buildings in red are the taller buildings. For all three methods, it is immediately clear that there are many similar building heights present. The reference model contains much higher variations in the building height for the same area. Which buildings have their height over- and underestimated is shown in Figure 5.3. In general, all three methods tend to over- and underestimate similar buildings. Only small differences are present, such as the connected building at the bottom of the map: RFR underestimated its height, while MLR and SVR both overestimated it.

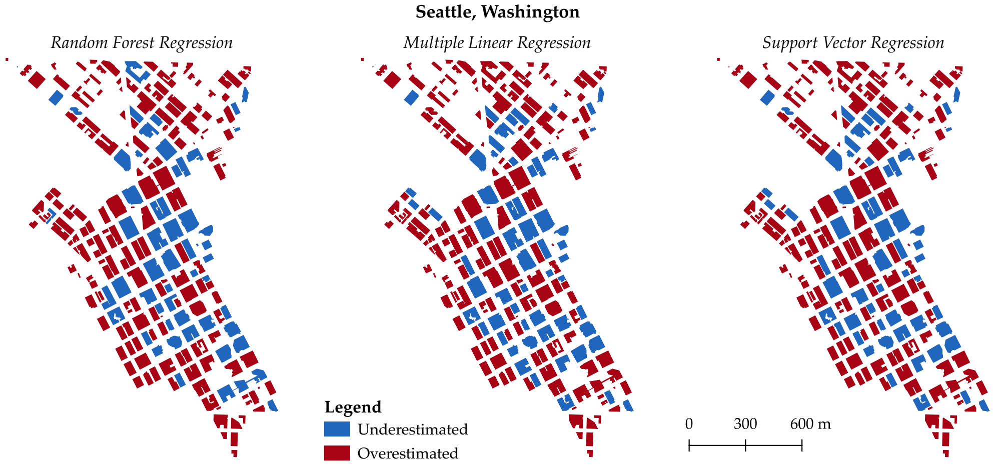


Figure 5.3: Maps showing the under- and overestimations of the building height predictions for the CBD of Seattle, Washington, for the three different machine learning methods. The prediction models are trained on only the CBD data.

Figure 5.4 shows the results of the three machine learning methods for a combined prediction model with the area morphology as an extra feature. The results of the RFR method look very similar to those in Figure 5.2, but the MLR and SVR methods changed significantly. Many more buildings are now coloured in blue, indicating that the predicted building heights are lower. The CBD model did struggle to produce such predictions. However, when using the combined prediction model, the MLR and SVR methods are less capable of predicting the heights of taller buildings (less yellow to orange buildings are present).

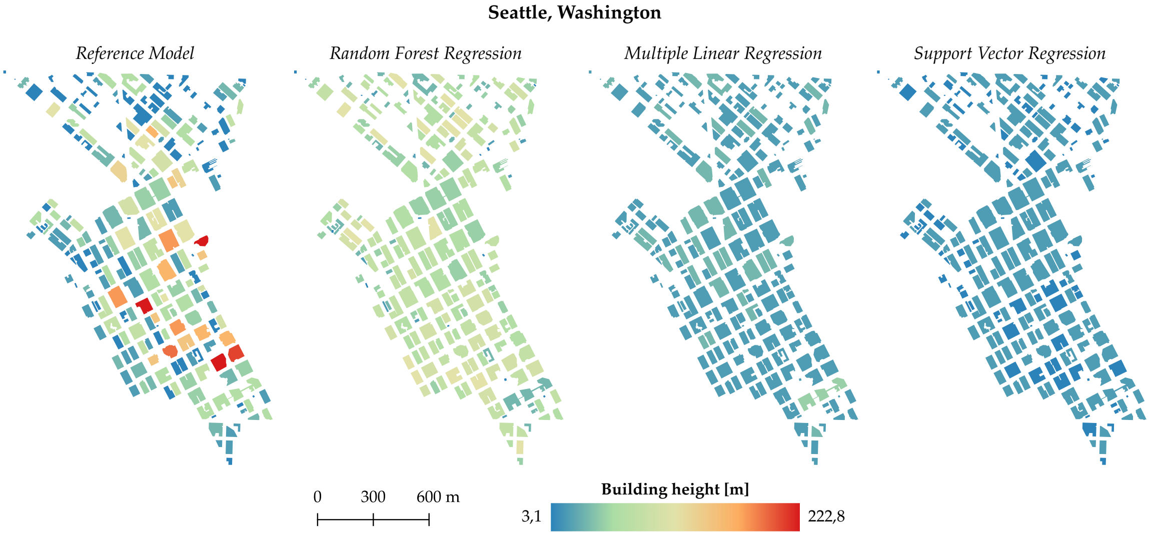
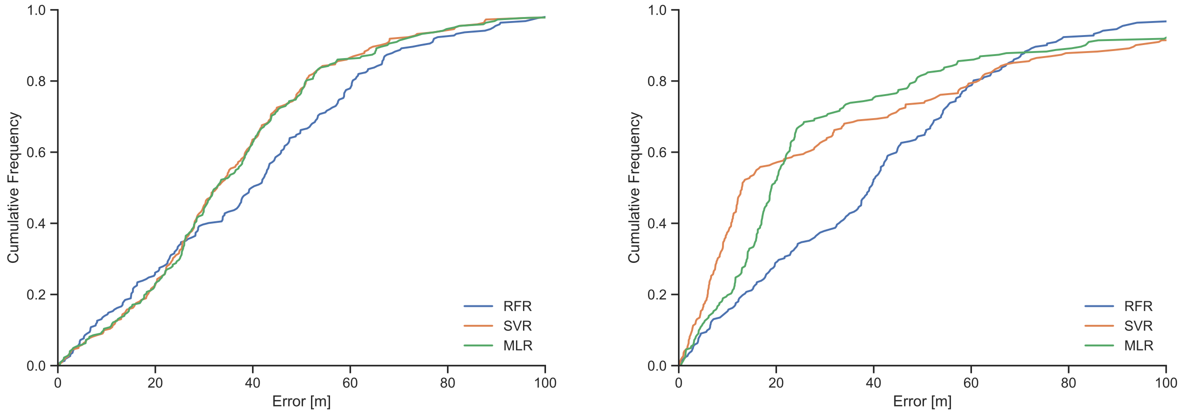


Figure 5.4: Maps showing a comparison between building heights in the reference model and the building height predictions of the three machine learning models for the CBD of Seattle, Washington. The prediction models are trained on all the training data with the area morphology as an extra feature.

Lastly, the Empirical Cumulative Distribution Function (ECDF) graphs in Figure 5.5 provide insight into the percentage of buildings in the dataset that express an error that is less or equal to x-metres. For the CBD prediction model (Figure 5.5a), the MLR and SVR methods show a similar trend: around 65% of the buildings in the dataset have an error of 40m or less. For the RFR method this number lies around 50%. Interestingly, for errors up to 20m the RFR method performs better than the other two, but then its performance drops. For the combined prediction model (Figure 5.5b), the SVR method does outperform the other methods for the lower building errors, but after 20m MLR performs best. Around 50% of the buildings have an error of 40m or less in the RFR predictions, while it is around 70% and 75% for the SVR and MLR methods respectively. Looking at the steepness of the graphs towards the 100m limit, we see that the RFR method is approaching 100% of the building footprints faster than the other two methods. This indicates that the two linear models have more buildings with extreme errors present.



(a) CBD prediction model

(b) Combined prediction model

Figure 5.5: ECDF for the three different machine learning methods for the CBD of Seattle, Washington. All nine geometric features are included in these predictions, and the combined model includes the area morphology. The graphs show the percentage of buildings having an error of x-metres or less.

5.2.2 Portland

The second test area is Portland, Oregon. According to the results in Table 5.3, the RFR method performs best when the suburban and rural prediction model is used. It achieves an MAE of 1.42m and a MAPE of 24.77%. An over- or underestimation of this magnitude is still relatively high, especially in the suburbs where the average building height tends to be lower. On a 10m high building, the error equals almost one entire storey. However, the error is still well within the 5m margin as is suggested in the CityGML specifications (see Section 3.5). For the MLR and SVR methods the MAE lies close to each other, with values of 1.67m and 1.65m respectively. The MLR method performs slightly worse with the combined prediction model, but the SVR method performs better. Its MAE of 1.41m is the lowest achieved error for the Portland test area.

Figure 5.6 provides a comparison between the reference model and the RFR method for the area around the city centre of Portland. It uses the suburban and rural prediction model. Because the results all lie so close to each other, only one method and model are shown. The reference model shows that there are many variations in the building height present in the city centre (left 'island' on the map). Comparing this area to the height predictions of the RFR method, it is clear that these variations are not present. In general, the RFR method expresses a lot less variation in the building height. In the suburban areas, the reference model also shows that many buildings are of similar heights. For these areas, the RFR method proves to work well.

A possible reason for the lower building heights around the city centre is that for Portland no CBD is detected. The suburban and rurally trained network is applied to the entire city. Since this network is trained mainly on shorter buildings, it fails to accurately predict the building height in areas with taller buildings. Figure 5.7 shows which building heights are under- and overestimated for the entire city. This map supports the initial visual inspection: around the city centre, the building heights are mainly underestimated. In the suburbs, however, there is a mix of under- and overestimations present.

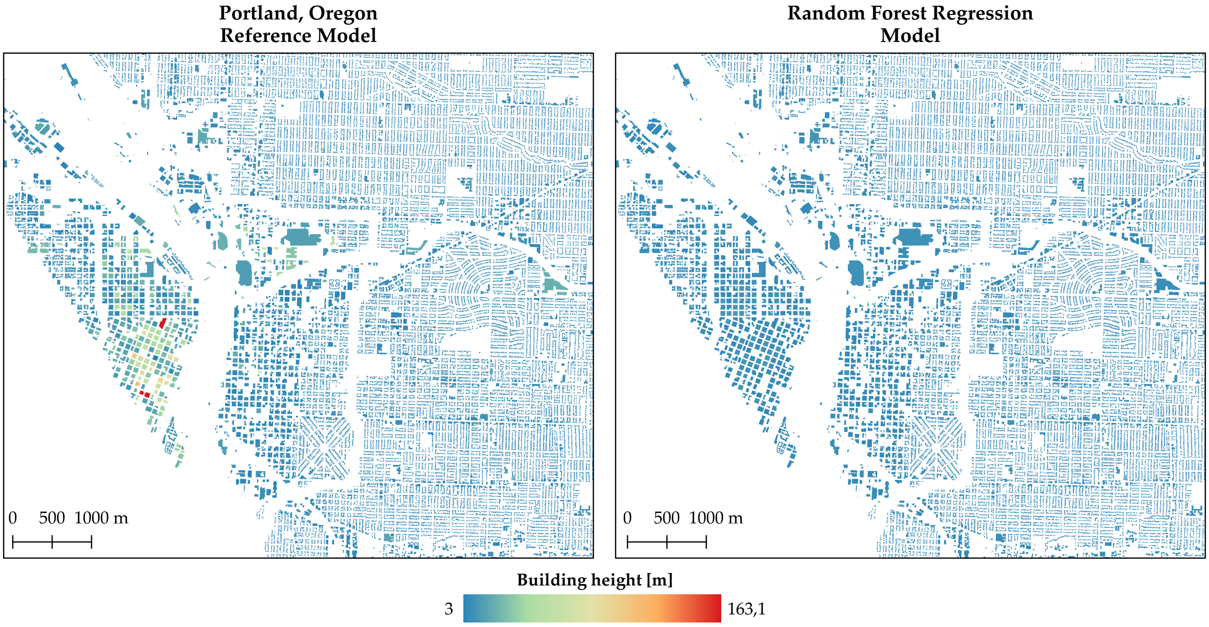


Figure 5.6: Maps showing a comparison between the building heights in the reference model and the RFR building height predictions for the area around the city centre of Portland, Oregon. The prediction model is trained on only the suburban and rural data. The RFR predictions contain a lot less variations in the building height than the reference model.

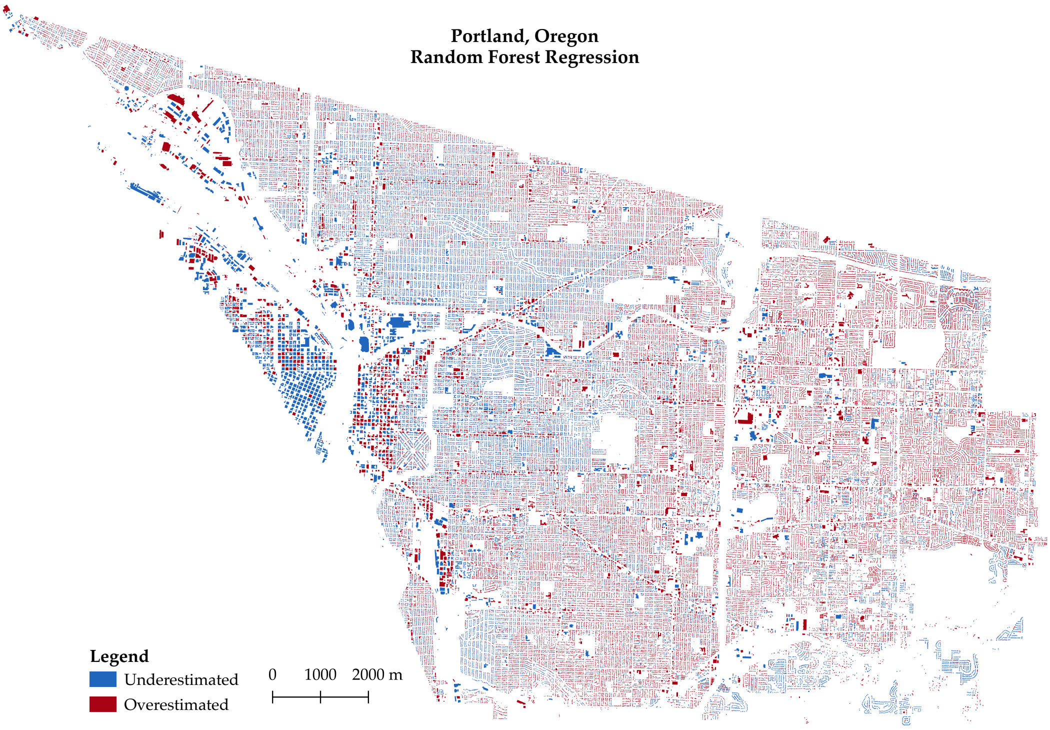
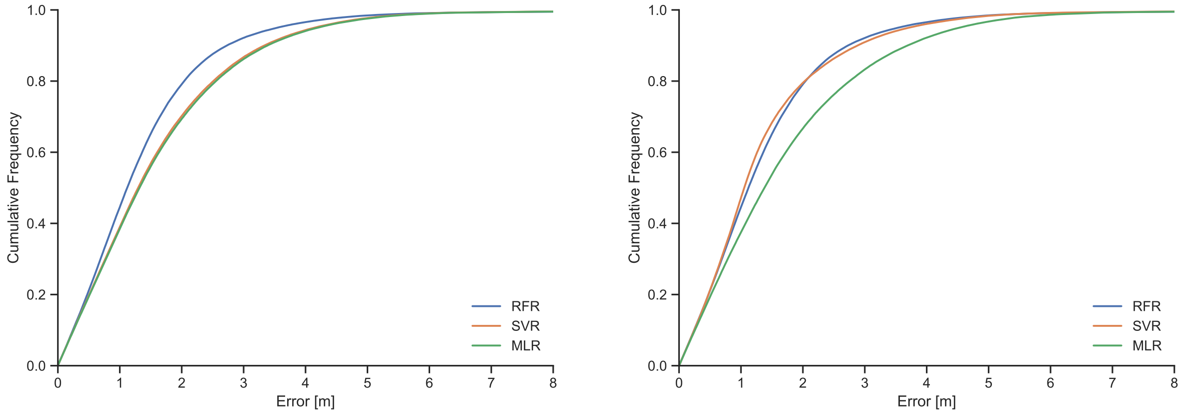


Figure 5.7: Map showing the under- and overestimations of the RFR building height predictions for Portland, Oregon. The prediction model is trained on only the suburban and rural data.

For the MLR and SVR methods, similar results as described above are found. To keep the structure clean and clear, those maps are not included in this section. All additional maps can be found in Appendix D.

The MLR and SVR methods are included in the ECDF graphs in Figure 5.8. Because the dataset contains many more samples than the Seattle dataset, the graphs are a lot smoother. For the suburban and rural prediction model (Figure 5.8a), the MLR and SVR methods follow a similar trend regarding the errors. For errors of 1m and less, all three methods perform quite similarly: around 45% of the buildings of the RFR method have an error of 1m or less, while for the MLR and SVR methods this number lies around 40%. After this, the slope of the graph for RFR is much steeper, indicating it has more buildings with smaller errors than the other two methods. The combined prediction model (Figure 5.8b) shows slightly different trends. Now, the RFR and the SVR methods perform more similarly. Both have around 45% of the buildings with an error of 1m or less. The performance of the MLR method has slightly dropped; now only around 35% of the buildings have an error of 1m or less.



(a) Suburban / Rural prediction model

(b) Combined prediction model

Figure 5.8: ECDF for the three different machine learning methods for the CBD of Portland, Oregon. All nine geometric features are included in these predictions, and the combined model includes the area morphology. The graphs show the percentage of buildings having an error of x-metres or less.

5.2.3 Astoria

The last test area is Astoria, Oregon. The results in Table 5.3 show that for the suburban and rural prediction network, all three machine learning methods have very similar results in terms of their MAE: 2.29m for RFR, 2.28m for MLR, and 2.27m for SVR. Again, like in the Portland dataset, the MAPE is still quite high (∼30%). This is especially important to consider because Astoria is a more rural region with generally lower building heights. For the combined prediction model, the results of the RFR and MLR methods are very similar to the predictions with the suburban and rural prediction model. The SVR method, however, performs worse in the combined model with an MAE of 2.51m.

Compared to Portland, the MAEs are higher for Astoria. The height predictions are performed with the exact same prediction models. A possible reason for this difference is that the dataset contains fewer buildings. The dataset of Portland contains about 35x more buildings than the dataset of Astoria. For Astoria, the averages are computed on much smaller sets, giving the outliers in the dataset a bigger influence on the results.

A visual comparison between the building heights in the reference model and those from the RFR method is shown in Figure 5.9. The suburban and rural prediction model is used to perform the predictions. Again, the combined prediction model is excluded because the results are very similar. From the maps, it is clear that, like in the Portland dataset, the model struggles to accurately perform predictions for buildings that express a higher building height in the reference model. Most buildings that have a yellow-red colour in the reference model are coloured blue-green for the RFR method. Figure 5.10 provides a better overview of which buildings are under- and overestimated. For the MLR and SVR methods similar results are found. The comparison maps are attached in Appendix D to keep this section uncluttered.

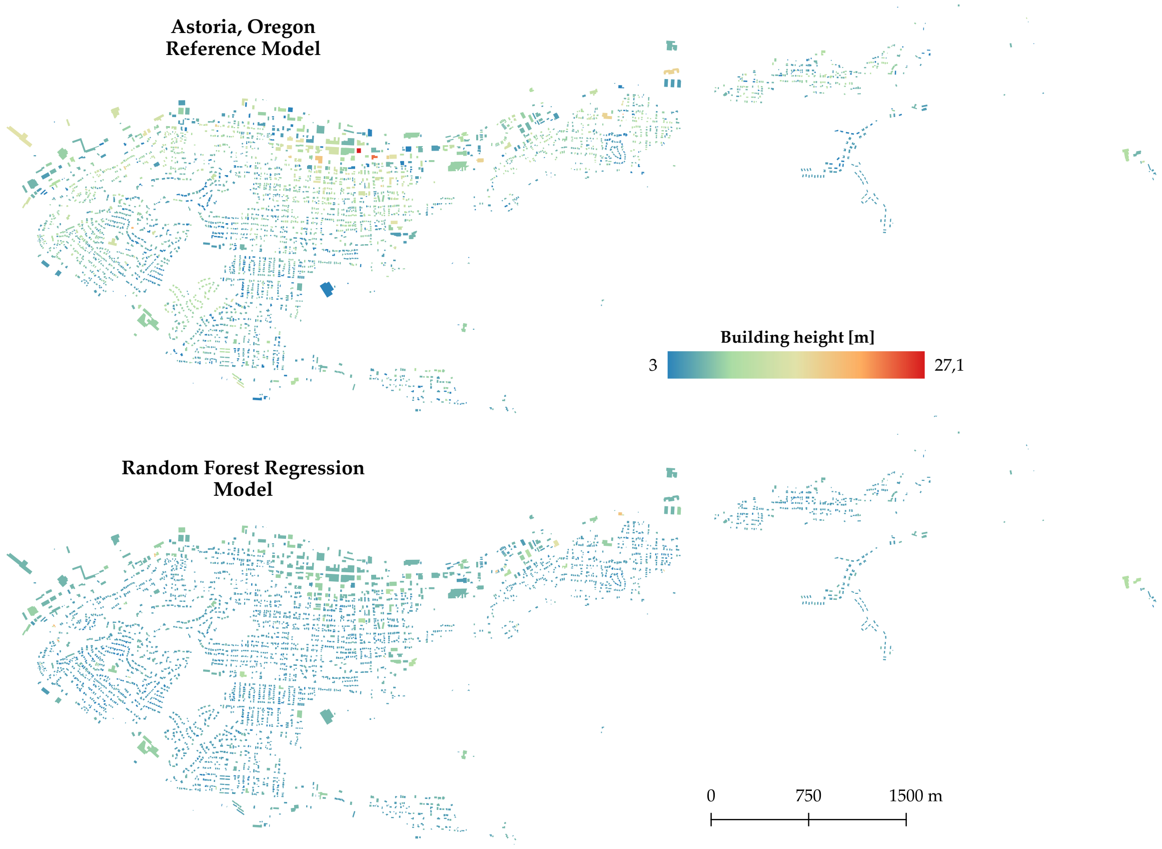


Figure 5.9: Maps showing a comparison between the building heights in the reference model and the RFR building height predictions for the town of Astoria, Oregon. The prediction model is trained on only the suburban and rural data.

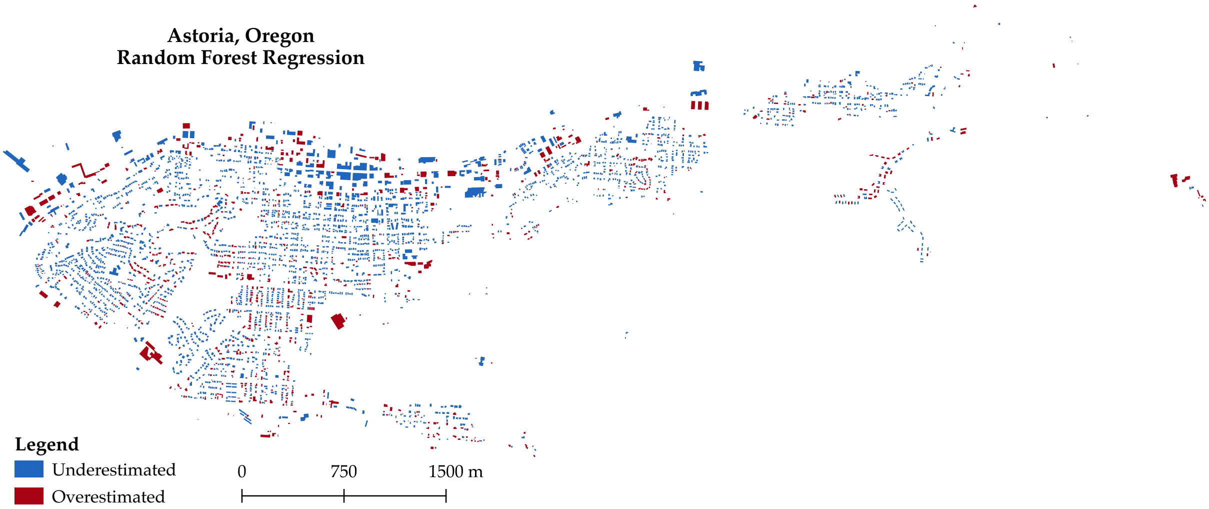
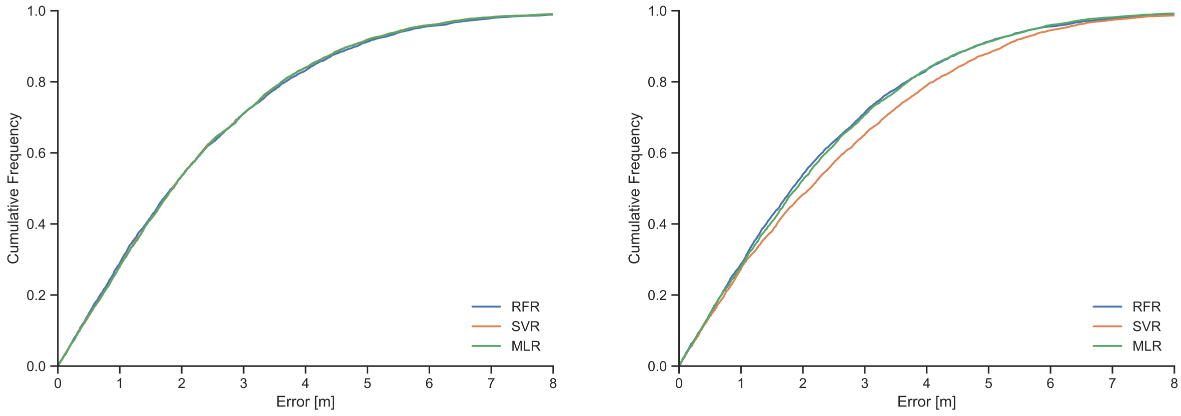


Figure 5.10: Map showing the under- and overestimations of the RFR building height predictions for the town of Astoria, Oregon. The prediction model is trained on only the suburban and rural data.

Lastly, the ECDF graphs in Figure 5.11 show the high similarity between the three machine learning methods. For the suburban and rural prediction model (Figure 5.11a), about 30% of the buildings in the dataset have an error of 1m or less, while around 90% of the buildings have an error of 5m or less. The combined prediction model (Figure 5.11b) shows similar results, but the performance of the SVR method slightly lags behind the other two methods. From these graphs, it is clear that many buildings in the Astoria dataset meet the CityGML recommendation for the maximum errors in an LOD1 model.



(a) Suburban / Rural prediction model

(b) Combined prediction model

Figure 5.11: ECDF for the three different machine learning methods for the CBD of Astoria, Oregon. All nine geometric features are included in these predictions, and the combined model includes the area morphology. The graphs show the percentage of buildings having an error of x-metres or less.

5.3 Omitting Area Morphology

Besides the combined prediction model that is trained with the area morphology as an extra feature, it is interesting to see how the prediction model performs if the area morphology is excluded. The results for this prediction model are presented in Table 5.4. As was the case with the prediction models that were discussed in the previous section, there is no single regressor that works best for all test scenarios.

City	Regressor	MAE [m]	MAPE [%]	RMSE [m]	RMSPE [%]
Seattle	RFR	36.62	61.66	57.14	70.65
	MLR	36.92	61.94	58.44	70.08
	SVR	37.06	61.87	58.54	69.70
Portland	RFR	1.63	28.69	3.08	49.89
	MLR	1.68	32.01	2.57	44.35
	SVR	1.66	31.43	2.54	43.29
Astoria	RFR	2.39	30.69	3.40	44.14
	MLR	2.15	30.21	2.79	40.82
	SVR	2.16	30.14	2.80	40.31

Table 5.4: Accuracy results for the three different test areas when the model is trained on all nine geometric features with only one prediction network. The area morphology is excluded as an extra feature.

Comparing these results to the errors in Table 5.3, we see that the MAE of the RFR method has dropped from 39.74m to 36.62m for the Seattle CBD. For the MLR and SVR methods, the combined prediction model with the area morphology included performs better. The RMSE did significantly increase for both the RFR and MLR method, indicating that the model fitted to the data might not be necessarily better. However, when looking at the percentage-based errors, both the MAPE and RMSPE are much lower when using the single prediction network without the area morphology. Figure 5.12 provides maps showing the building height predictions of the three regressors based on this single prediction network next to the height values in the reference model. Compared to Figure 5.4, the height estimations for the RFR method are significantly lower using the model without the area morphology. This could also explain why the MAE is lower because the model is better at predicting the heights for the shorter buildings in the dataset. For the MLR and SVR methods the height distributions look fairly similar, but the MLR method with the area morphology included in the prediction model seems to capture taller buildings better.

For this comparison, it must be considered that Seattle is a small test area with only 223 buildings. Other CBDs in the USA might express different characteristics, resulting in different results for when the different prediction models are compared.

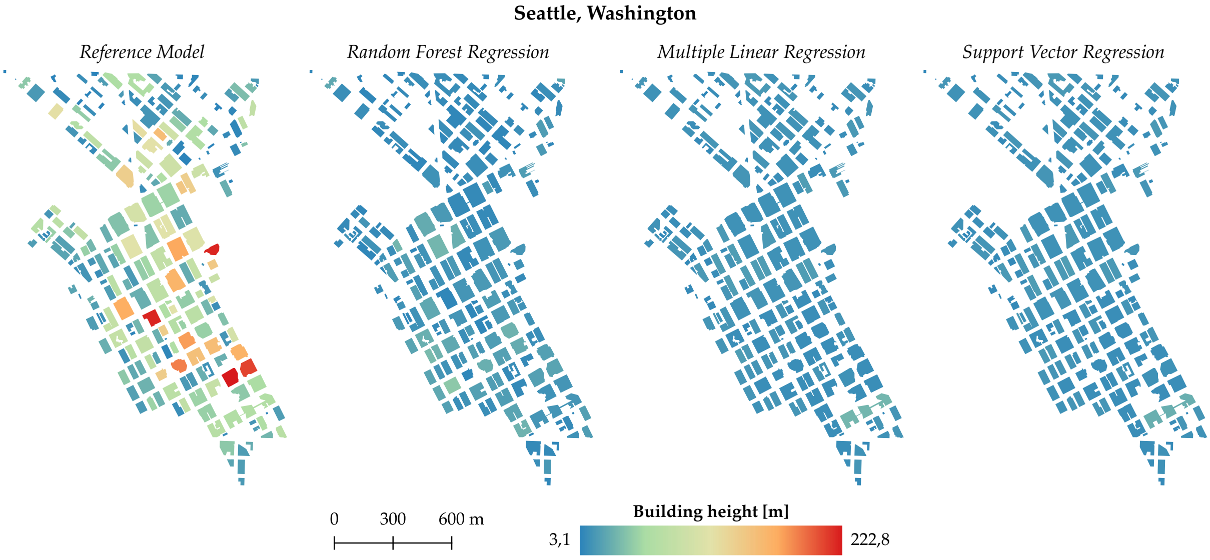


Figure 5.12: Maps showing a comparison between building heights in the reference model and the building height predictions of the three machine learning methods when one prediction network is used for the CBD of Seattle, Washington. The area morphology is excluded as an extra feature.

When looking at the results for Portland, the MAE did increase for the RFR and the SVR methods when the morphology is excluded. However, for the MLR method the approach without the area morphology works better than the one that includes it as a feature. But, the rural and suburban prediction model performs even slightly better than the model presented in this section (MAE of 1.67m vs. 1.68m). Also, for all three machine learning methods, the percentage-based errors did increase compared to the results presented in Table 5.3. This means that the error relative to the true building height became larger. The prediction model that excludes the area morphology is thus not the best fit for the city of Portland.

Lastly, for Astoria the results are mixed. While the RFR method performs worse with the prediction model without the area morphology, both the MLR and SVR methods improved in terms of their MAE. For the MLR method, the MAE decreased from 2.28m to 2.15m, while for the SVR method it went down from 2.27m to 2.16m. However, like for Portland, the percentage-based errors did increase for all three methods.

This comparison makes clear that selecting a proper prediction model is not as straightforward as it might sound. Depending on the test area and the machine learning method, different approaches work best. The results of the prediction models heavily depend on the data that is used to train the models. Using a wider variety of CBD training data could possibly improve the results of the models. Also, using different feature subsets — to better capture the building characteristics — could also improve the results (see Section 5.6).

5.4 Comparison Open City Model

In this section, the results of the three machine learning methods are compared to the building heights present in the OCM. The comparison is like comparing apples to oranges because there are many differences between the OCM and the models presented in this thesis. For starters, the OCM combines building height information from several datasets. Especially in the bigger cities, the building heights are often not estimated using machine learning. In this thesis, I try to predict the building heights for all buildings in the conterminous USA and try to consider the different area morphologies that are present. Second, it is also not entirely clear which features and machine learning methods are used by the OCM to predict the building heights. The developers state that they use some simple regression analysis [BuildZero, 2019], but whether this equals the linear regression implementation in this thesis project is unclear. They might also use fewer, or even completely different, features than I do to build the models. Third, I do not know exactly what data is used for training the prediction model(s) of the OCM, making it difficult to draw conclusions because we cannot compare the characteristics of the training sets. Lastly, I only consider buildings that have a height greater or equal to 3m in the training process. For the OCM it is unclear whether a similar decision is made, or if it includes all building height values in a dataset.

City	MAE [m]	MAPE [%]	RMSE [m]	RMSPE [%]
Seattle	11.04	62.58	21.30	149.50
Portland	1.20	22.13	2.32	39.12
Astoria	2.37	29.03	3.04	33.83
Astoria (filtered)*	2.16	27.29	2.76	31.56

* Only the footprints that are present in the `USBuildingFootprints` dataset are included

Table 5.5: Accuracy results based on the height values present in the OCM datasets.

Even though all these differences are present, I do still perform a comparison because it is interesting to see how the machine learning models compare to a different model. For each OCM dataset, a reference model is created using the same LiDAR data as was used for the three test areas described in the Section 5.2. Only the buildings with a ground truth height of 3m or higher are included in the comparisons. The accuracy results of the building heights in the OCM are presented in Table 5.5. For Astoria, two entries are included. The first is based on all footprints in the OCM, which also includes footprints from OpenStreetMap (OSM). The latter only includes the footprints that are from the `USBuildingFootprints` dataset, which are the building footprints that are used in this thesis. By only including these footprints, a fair comparison can be made between the results of the OCM and the models presented in this thesis. For Seattle and Portland, this division is not made. These areas highlight a problem that is present for many cities in the USA in the OCM: the footprints are either not from the `USBuildingFootprints` dataset, or the heights are not estimated using machine learning. Finding a city that has the correct building footprints, uses machine learning for the building heights, and has reference data available is a difficult task. The city centre of San Diego, of which a short data quality analysis is provided in Section 4.1, does have the building heights estimated and has reference data available. However, it does not make use of the `USBuildingFootprints` dataset and it, therefore, is excluded from this section as an extra test area.

We will first consider the results for Seattle and Portland. Both these areas do not have the exact same building footprints available as used in the models in this thesis. These two comparisons are included to see how well the other building height sources of the OCM match the heights in the reference model. Table 5.5 shows that the OCM has an MAE of 11.04m for Seattle, Washington. This is significantly lower than the MAEs for the combined prediction model (see Table 5.3), which vary from 32.84m to 39.74m A breakdown of the OCM data for the Seattle CBD is provided in Figure 5.13. The map on the left indicates the different height sources that are used, and it is clear that only a few building heights are estimated using machine learning. The two maps in the middle compare the heights in the reference model to those in the OCM. Visually, they look similar. The OCM provides a better real-world scenario than the three machine learning methods shown in Figure 5.2 and 5.4. However, an MAE of 11.04m is still significant as it is more than three building storeys (assuming 3m for one storey). When inspecting the differences for only the buildings that get their height from the OCM, we see that those buildings do almost all deviate quite significantly from the ground truth. Often, the heights are underestimated as is shown in the right-most map of Figure 5.13.

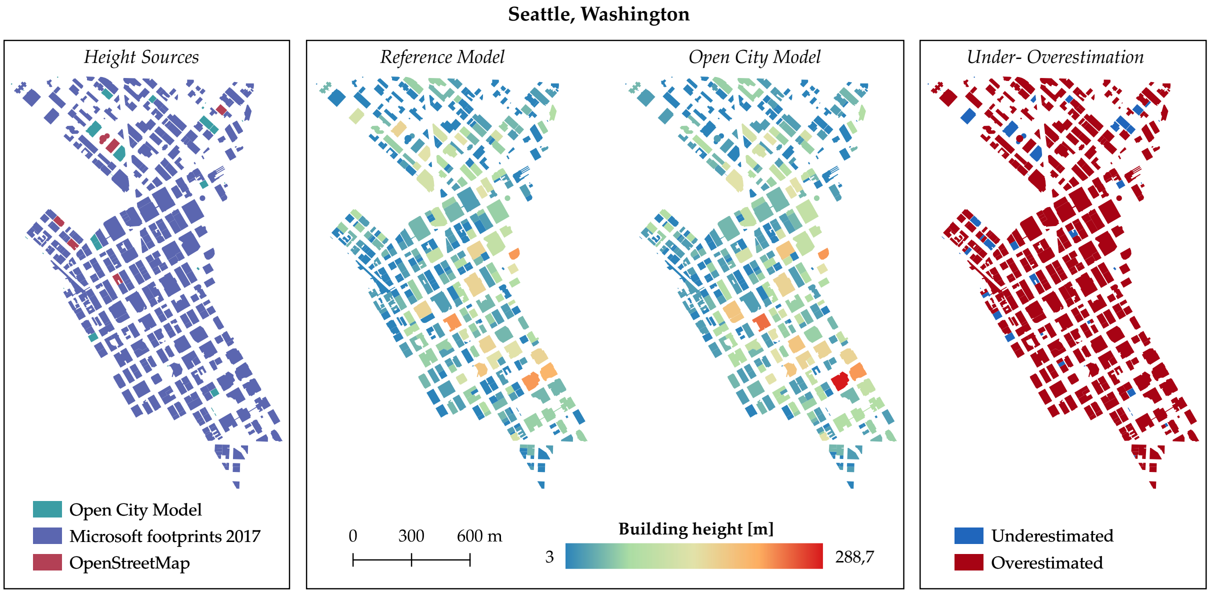


Figure 5.13: Comparison of the building heights in the OCM and a reference model for Seattle, Washington. The map on the left shows that most buildings in the Seattle CBD get their building height from datasets other than the OCM. The map on the right shows the under- and overestimations based on the reference model.

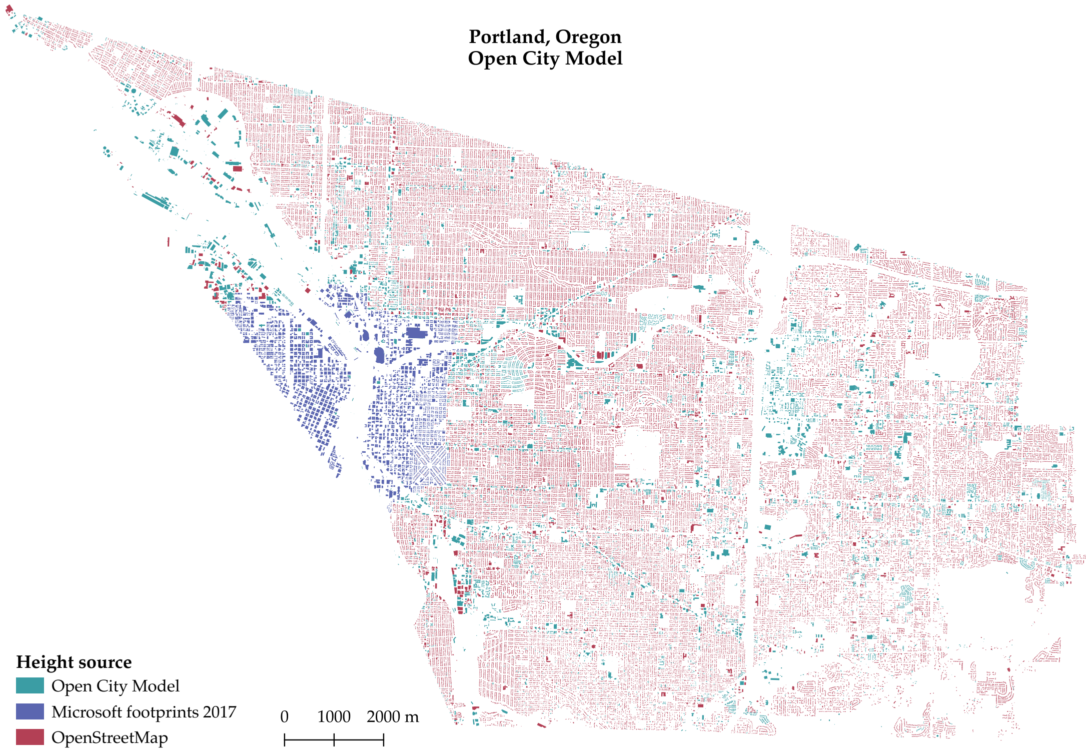


Figure 5.14: The height sources for the building footprints in the OCM dataset for Portland, Oregon. Building heights around the city centre and in the suburbs are mainly from other datasets than the OCM.

For the city of Portland, Oregon, we find similar trends. Many of the building heights in the city centre and the suburbs are derived from other data sources (see Figure 5.14). Scarcely scattered around the city, we do see several areas where the OCM enriched the building footprints with estimated height values. Considering all building footprints, the Portland OCM has an MAE of 1.20m. The three machine learning methods from this thesis perform slightly worse, with a lowest MAE of 1.41m for the SVR method that uses the combined prediction model (see Table 5.3). However, this error is still quite low, especially when considering that all the building heights are estimated. A close-up of the area around the Portland city centre is shown in Figure 5.15. The building heights in the city centre are more accurately portrayed in the OCM than in the RFR model from Figure 5.6. In the suburbs, however, the results look similar for both models.

Appendix D contains extra maps, including the under- and overestimations of the OCM for the whole city and a close-up of the Portland city centre.

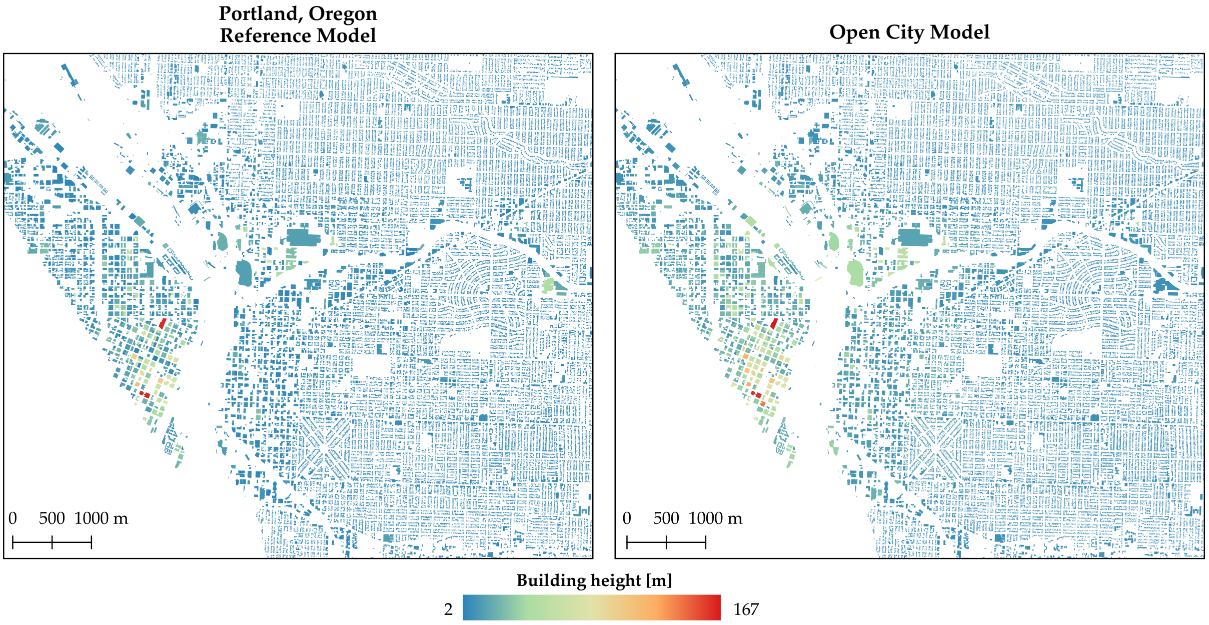
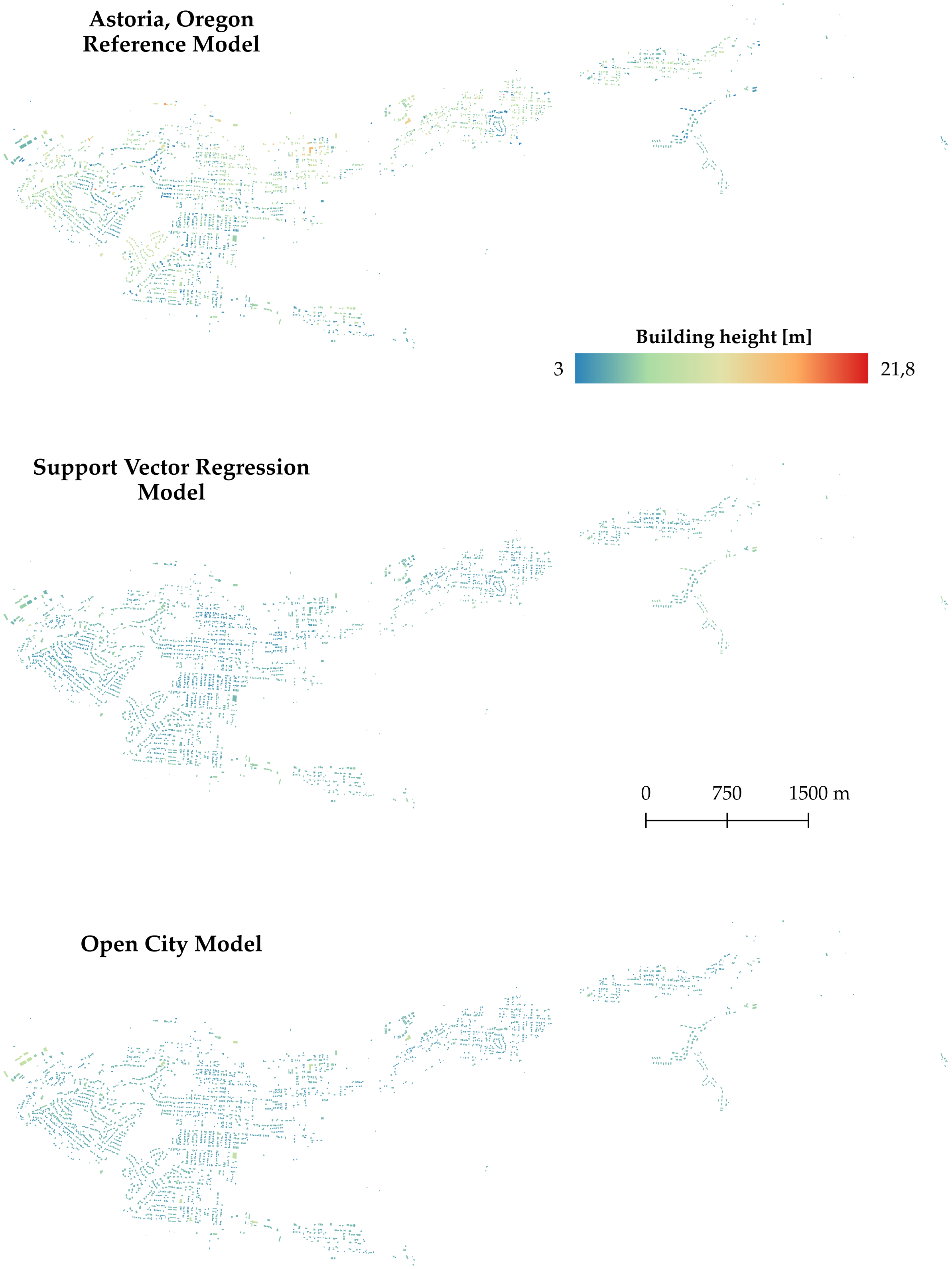


Figure 5.15: Comparison of the building heights in the OCM and a reference model for the area around the city centre of Portland, Oregon.

The last comparison is for Astoria, Oregon. The original Astoria dataset contains 3,723 buildings (see Table 4.1), but after removing the footprints that are different in the OCM only 2,913 footprints remain. The accuracy of the height predictions for these footprints — based on my prediction models — are shown in Table 5.6. The results are split up between a prediction model that is trained with only the suburban and rural data, and a model that is trained on all data but uses the area morphology as an extra feature. Comparing these results to the prediction errors of the OCM defined in Table 5.5, the prediction models from this thesis do, in general, provide better results. The MAE of the OCM is 2.16m, and the lowest achieved MAE in this thesis is 2.09m for the MLR and SVR methods using the prediction model trained on only the rural and suburban data. In terms of the other error metrics, the SVR method performs slightly better than the MLR method.

A visual representation of the results is shown in Figure 5.16. It includes the reference model for the area, and the best SVR model and the OCM. The reference model contains more variations in the building height than the other two models; there are more taller buildings present. The maps also show that the SVR method tends to estimate lower building heights for bigger building footprints than the OCM. But, other than this, the results look very similar. Maps for the RFR and MLR methods are attached in Appendix D.


Astoria, Oregon
Reference Model

Support Vector Regression
Model

Open City Model

Figure 5.16: Comparison of the building heights in the reference, SVR, and OCM model for the town of Astoria, Oregon. The SVR height predictions are performed using a prediction network trained on only rural and suburban data.

Regressor		Prediction model	
		Suburban / Rural	*Combined*
RFR	*MAE [m]*	2.11	2.11
	MAPE [%]	27.37	27.41
	RMSE [m]	2.73	2.73
	RMSPE [%]	33.38	33.34
MLR	*MAE [m]*	2.09	2.11
	MAPE [%]	28.09	28.40
	RMSE [m]	2.65	2.70
	RMSPE [%]	33.78	34.27
SVR	*MAE [m]*	2.09	2.33
	MAPE [%]	27.89	29.07
	RMSE [m]	2.64	2.95
	RMSPE [%]	33.34	32.97

Table 5.6: Accuracy of the building height predictions for Astoria, Oregon, for two different types of prediction models. The models use all available geometric features, and the combined model adds the area morphology as an extra feature. Only the building footprints that are present in the OCM are included in the tests.

5.5 Impact of Non-Geometric Features

In addition to the test with the geometric features, the city of Denver is used to test how additional non-geometric features affect the height prediction results. 25% Of the Denver dataset is used for training the prediction model(s), while 75% is used for performing the height predictions. Table 5.7 shows the results of these tests. For the CBD and the suburban results, two models were trained separately based on the area morphology. The combined model does not make this distinction, but it does include the area morphology as an extra feature in the *base*-model. For the CBDs and the suburban data, the *base*-model only contains the nine geometric features. In all three test situations, the *enriched*-model adds the following features to each of the building footprints:

1. *Building type*. The type of the building footprint, which could be any of the following as defined in the Denver dataset: `garage/shed`, `residential`, `commercial`, `industrial`, `misc`, `public`, `medical`, `parking structure`, `tank` and `foundation/ruin`.
2. *Average household income*. The average household income defined based on the census tracts (see Figure 4.7a). No clear pattern is present for the suburbs and the CBD in terms of average household income.
3. *Average household size*. The average number of people that form a household. This value is defined based on the census tracts (see Figure 4.7b). In the suburbs the average household size tends to be bigger than in the city centre.
4. *Population density*. The population density in people / sq. mile defined based on the census tracts (see Figure 4.7c). In the city centre the population density tends to be higher than in the suburbs.
5. *Number of amenities*. For each building the amenities within a 250m radius of the building centroid are counted. These amenities are based on data available in OSM.
6. *Raster building height indication*. The average height for a building derived from the raster cells of an underlying DEM that contains the heights relative to the terrain.

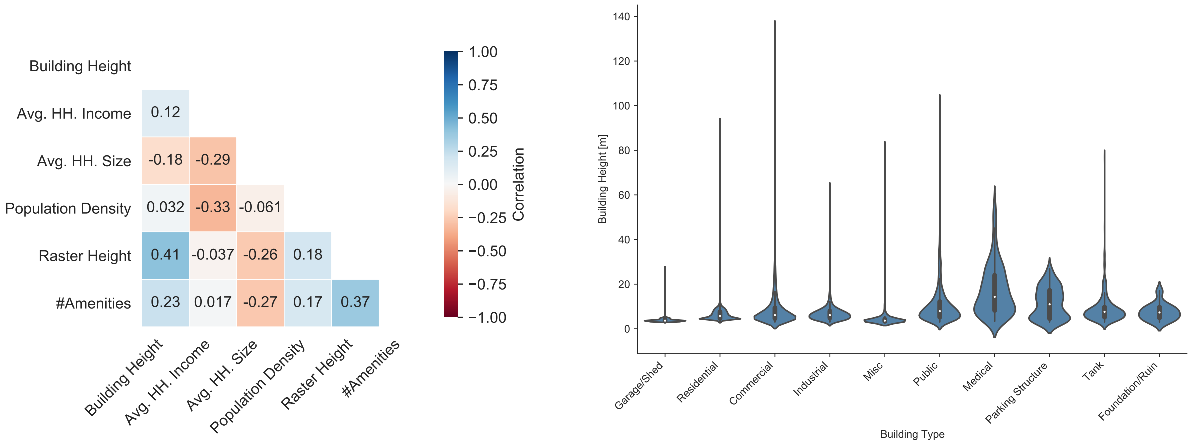
Looking at the results in Table 5.7, we see that adding the non-geometric features is beneficial for all three machine learning methods. With the suburban prediction model, the RFR method reduces its errors the most and it achieves sub-metre accuracy with an MAE of 0.96m. The MLR method has the second-highest reduction in its MAE, followed by the SVR method. A possible reason why the RFR method benefits most from the new features could be its ability to detect non-linear relationships in the data. The MLR and

SVR methods do not have this capability. The non-geometric features do not necessarily express a linear relationship to the building height. Especially because some of the features are assigned to the building footprints based on the census tracts. These areas can contain a large variety of buildings with different heights, and short and tall building are then assigned the same census tract values.

		Suburbs		**CBDs**		**Combined**	
Regressor		*Base*	*Enriched*	*Base*	*Enriched*	*Base*	*Enriched*
RFR	*MAE [m]*	1.35	0.96	20.84	17.29	1.44	1.03
	MAPE [%]	22.05	15.68	152.40	114.66	22.67	16.02
	RMSE [m]	2.71	2.11	30.68	27.12	3.55	2.93
	RMSPE [%]	33.30	25.08	267.17	208.19	36.97	28.52
MLR	*MAE [m]*	1.59	1.47	21.33	16.87	1.72	1.60
	MAPE [%]	26.81	25.16	158.71	109.09	27.99	26.66
	RMSE [m]	2.93	2.58	31.55	28.57	3.80	3.37
	RMSPE [%]	35.05	33.73	246.51	200.89	40.21	37.75
SVR	*MAE [m]*	1.55	1.46	26.10	25.80	1.68	1.59
	MAPE [%]	23.94	23.79	87.49	89.80	25.12	24.93
	RMSE [m]	2.98	2.64	41.45	39.83	3.84	3.41
	RMSPE [%]	31.21	31.07	88.19	95.24	36.17	34.84

Table 5.7: Results of the three different machine learning methods for models trained on only geometric features (*base*) and models enriched with non-geometric features for Denver, Colorado. All nine geometric features are included, and in the case of the combined model the morphology is added in the *base*-model. 25% Of the dataset is used for training the model, while 75% is used for testing.

Figure 5.17a shows Pearson's correlation coefficient for the suburban data for the non-geometric features that are of numerical nature. It shows that the population density feature is almost negligible for the linear models; the coefficient is almost zero. The average household income and the number of amenities have a low positive correlation to the building height, while the average household size expresses a low negative correlation. This shows that these features have no strong linear relationship to the building height. With a correlation value of 0.41, the building height estimate derived from the raster expresses a relatively high positive correlation to the true building height.

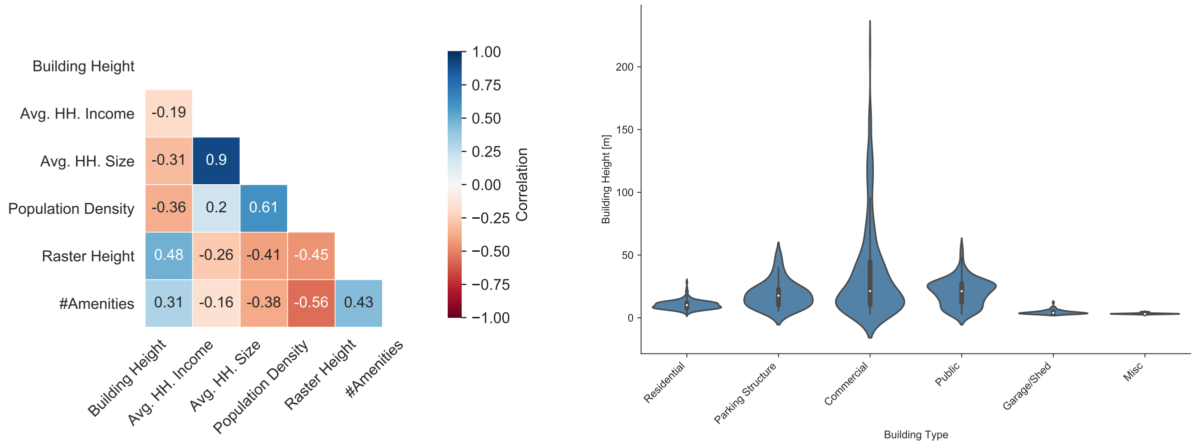


(a) Pearson's correlation coefficient

(b) Violin plot

Figure 5.17: The relation between the non-geometric features and the building height for the Denver suburban areas. The white dot in (b) indicates the median height value for that class.

For the relation between the building types and the building height violin plots can be used. Figure 5.17b shows the distribution of the building height for each building type that is present in the suburbs. The white dot in the violin plot indicates the median height for that building type, and all violins are scaled to be of equal width. The `parking structures` and `medical` buildings stand out the most in terms of their violin plot shape and their median height value that lies higher than for the other classes. All other classes express relatively similar violin shapes, possibly because of the high number of building types. From these building types, the `garage/shed` buildings express the lowest building heights. Most other classes include taller buildings, causing the median height to be roughly the same for all of them.



(a) Pearson's correlation coefficient

(b) Violin plot

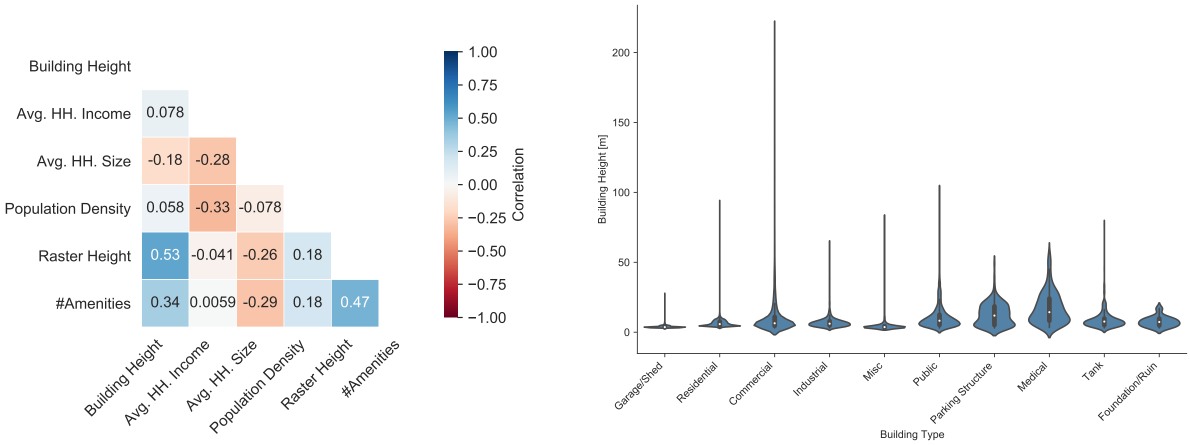
Figure 5.18: The relation between the non-geometric features and the building height for the Denver CBD. The white dot in (b) indicates the median height value for that class.

For the Denver CBD, the MLR method benefits the most from adding the non-geometric features and it reduces its MAE by 4.46m. The RFR method follows closely, with an improvement of 3.55m. Again, the SVR method improved the least with only 0.3m. It is interesting that for this case, the linear methods are affected differently by adding the new features. The Pearson's correlation coefficients in Figure 5.18a express the linear relationships between the features. The average household income has a low negative correlation to the building height, while the average household size and the population density express a higher negative correlation. The number of amenities and the height derived from the raster both express a positive correlation to the building height. The raster height has the highest correlation coefficient, as was the case for the suburbs. The linear relationships between the non-geometric features and the building height are clearly more present than in the suburbs, giving a possible explanation of why the MLR method can benefit more. It does not provide an explanation for the limited improvement of the SVR method.

Looking at the relation between the building height and the building types in Figure 5.18b, shows that `commercial` buildings express the highest building heights, while `garage/shed` and `misc` express the lowest. The `parking structures` and `public` buildings are somewhere in the middle. These differences between the classes should make it possible for a machine to discriminate between them.

The last analysis is for the combined prediction model that is trained with both suburban and CBD data. It uses the area morphology as an extra feature in the training and prediction stages. The results for the combined model show similar trends to the results for the suburbs: the RFR method greatly benefits from adding the new features, but the MLR and SVR methods do not show any significant improvements. The Pearson's correlation coefficients in Figure 5.19a also show similarities to those of the suburban regions (see Figure 5.17a). The average household income and population density have a correlation coefficient of almost zero, indicating that no linear relationship was found to the building height. The average household size still expresses a low negative correlation to the building height, and the number of amenities and the raster building height values prove to have the highest linear relationship to the building height. The less clear linear relationships for some of the features could again explain why the MLR and SVR methods show fewer improvements in terms of their MAE when the new features are included.

Figure 5.19b shows the relation between the building types and the building height in a violin plot. The trends are again very similar to those present in the suburbs in Figure 5.17b, but the maximum height changed because of the taller buildings that are present in the CBD. As with the suburbs, the distinction between the building types could possibly be difficult.



(a) Pearson's correlation coefficient

(b) Violin plot

Figure 5.19: The relation between the non-geometric features and the building height for the combined area morphologies in Denver. The white dot in (b) indicates the median height value for that class.

5.6 Feature Subsets

As described in Section 4.4, selecting an optimal subset of features is a challenging process. The multicollinearity among the geometric features makes the process even more difficult. But since we are dealing with the entire USA, it is preferred that we use the geometric features because they are always available. The three different machine learning methods are another complicating factor. For the MLR and SVR methods linear relationships are preferred, meaning that the Pearson's correlation coefficient should be high. The RFR method, on the other hand, can also detect non-linear relationships.

	Feature								
#	A	Cp	N	Co	Adj	L	W	S	V
1.	x								
2.	x		x						
3.			x				x		
4.	x		x	x		x	x		
5.	x	x	x			x	x		
6.	x		x					x	
7.	x		x						x
8.	x		x		x				
9.					x			x	x
10.	x	x	x	x	x	x	x	x	x

(a) CBDs

	Feature								
#	A	Cp	N	Co	Adj	L	W	S	V
1.					x				
2.	x								
3.	x				x				
4.		x		x	x	x			x
5.	x		x	x	x		x		
6.	x				x			x	
7.	x				x				x
8.								x	x
9.	x	x	x	x	x	x	x	x	x

(b) Suburban / Rural areas

Table 5.8: Feature subsets for the two separate area morphologies. Legend: A – area, Cp – compactness, N – #neighbours, Co – complexity, Adj – #adjacent buildings, L – length, W – width, S – slimness, V – #vertices.

Based on the findings of Section 4.4, subsets of features are defined for the two area morphologies and for the training set that combines all data. Table 5.8a shows the subsets that will be tested for the CBDs.

Model 1 considers only the area, which is the most important feature according to the permutation importance in the RF. Model 2 adds the number of neighbours to this model, which is the third most important feature. Then, the number of neighbours and width are combined in model 3. The area is left out in this model because of its collinearity with the building width. Next, model 4 combines the top five features based on Pearson's correlation values. Multicollinearity is present among these features as shown in Table 4.7. Similarly, model 5 combines the top five features from the permutation importance in the RF. The three following models are selected to study the impact of features that express a low feature importance. The area and number of neighbours from model 2 are taken as a base, where model 6 adds the slimness, model 7 the number of vertices, and model 8 the number of adjacent buildings. Lastly, model 9 only includes the three least important features, and model 10 simply uses all available features.

A similar approach is taken for the suburban and rural areas, and the selected subsets are shown in Table 5.8b. Model 1 only includes the number of adjacent buildings, which is the most important feature in both the permutation importance and the correlation analysis. Model 2 considers only the area, which is the second most important feature according to the permutation importance in the RF. These two features are then combined in model 3. Next, model 4 combines the top five features based on Pearson's correlation values. Like with the subset of the CBDs, multicollinearity is present. Model 5 combines the top five features from the permutation importance in the RF. The last three models include the two least important features to study their impact. Model 6 and 7 use the area and number of adjacent buildings in combination with the building slimness and number of vertices respectively. Lastly, model 8 only includes the building slimness and the number of vertices, and model 9 includes all available geometric features.

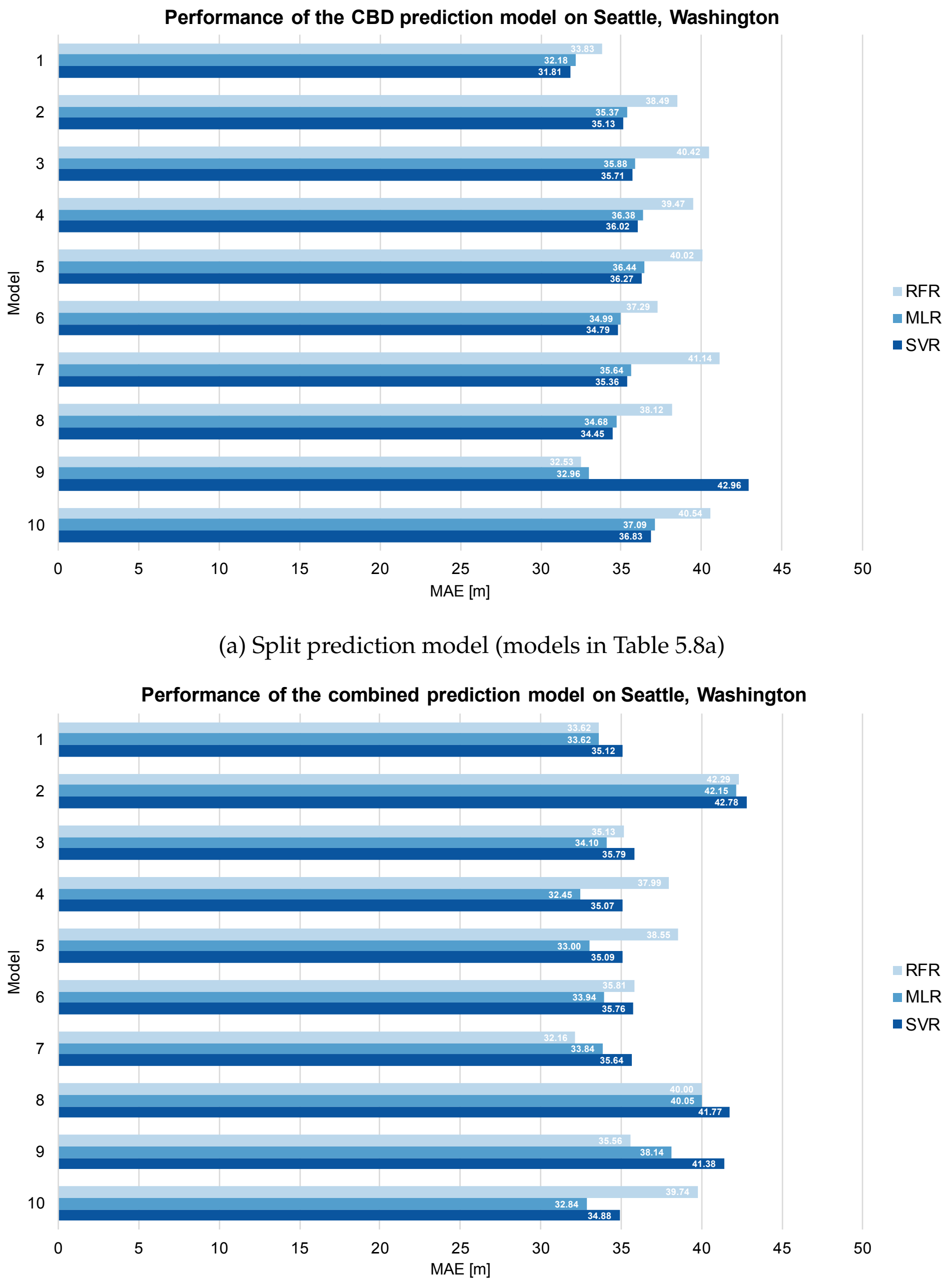
	Feature									
Model nr.	A	Cp	N	Co	Adj	L	W	S	V	M
1.										x
2.					x					
3.					x					x
4.				x	x	x	x			x
5.	x		x		x	x				x
6.					x			x		x
7.					x				x	x
8.								x	x	
9.	x									
10.	x	x	x	x	x	x	x	x	x	x

Legend: A – area, Cp – compactness, N – #neighbours, Co – complexity, Adj – #adjacent buildings, L – length, W – width, S – slimness, V – #vertices, M – morphology

Table 5.9: Subsets of features for the combined training dataset.

Lastly, the subsets of features for the training dataset that combines all area morphologies are shown in Table 5.9. Model 1 only includes the morphology; the most important feature in both the correlation and permutation importance (see Table 4.9). Model 2 contains only the number of adjacent buildings, which is the second most important feature. The two most important features are then combined in model 3. Model 4 includes the top five features based on Pearson's correlation values, and model 5 combines the top five of the permutation importance. Models 6, 7, and 8 include the least important features; the slimness and number of vertices. The first two combine these features with the two most important features (model 2), and in model 8, they are tested together. Model 9 is added to check if the building area would be sufficient to predict the building heights, and model 10 does again include all possible features.

The results of the feature subset tests on the test areas of Seattle, Portland and Astoria are shown in Figure 5.20, 5.21 and 5.22 respectively. The figures include the MAE for the different prediction models for the three machine learning methods, where a lower MAE indicates better model performance. The full results, including the RMSE, MAPE and RMSPE are attached in Section D.3.



(a) Split prediction model (models in Table 5.8a)

(b) Combined prediction model (models in Table 5.9)

Figure 5.20: Results of the feature subsets for the three machine learning methods tested on Seattle, Washington.

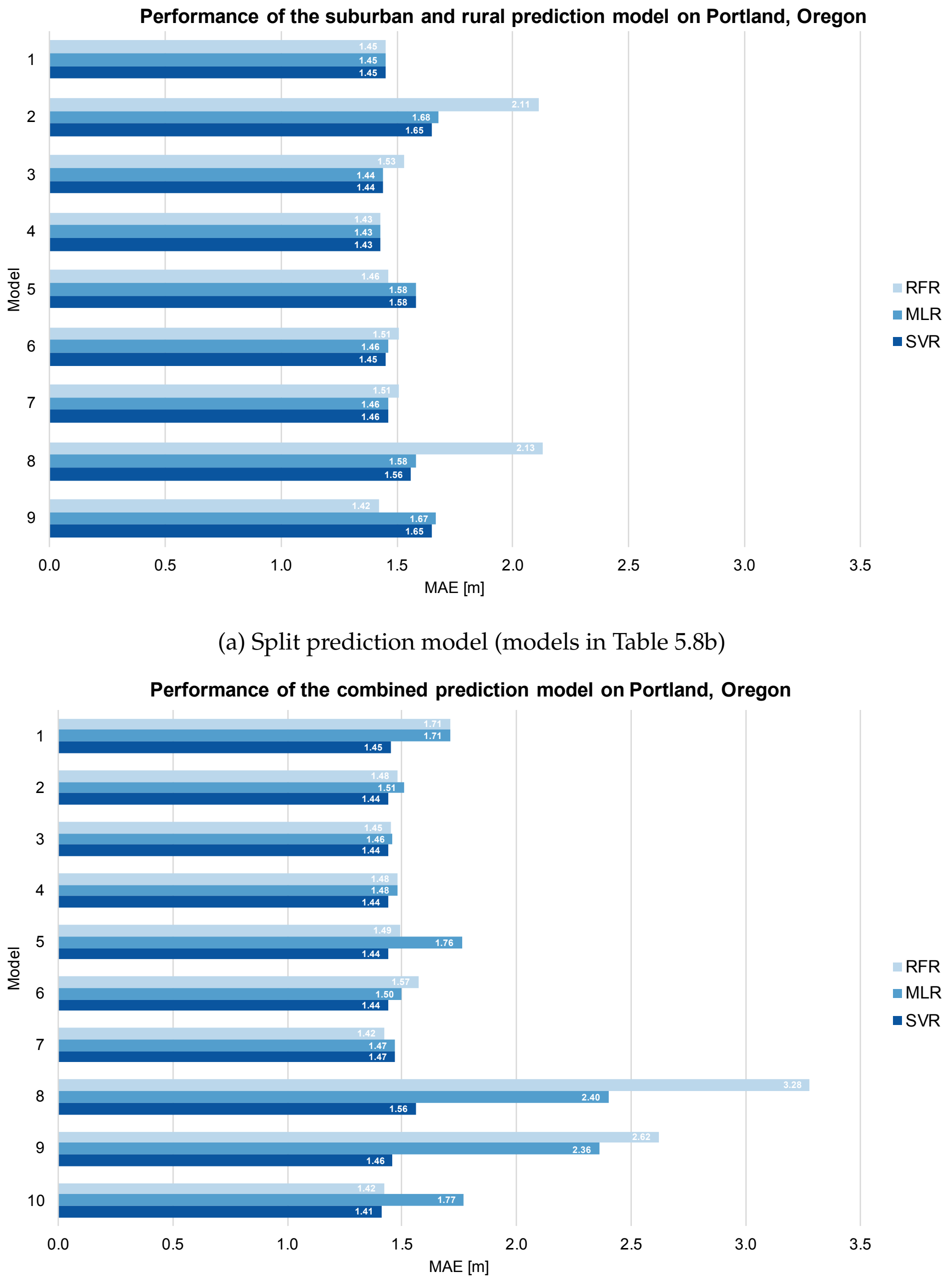
First of all, we will look at the performance of the prediction model — trained with only CBD data — on the Seattle test area (see Figure 5.20a). The performance of the RFR, MLR and SVR methods clearly differs for the different feature subsets. The RFR method has the worst performance in all cases, except in model 9 where it outperforms the other two methods. Model 9 is also the optimal subset for the RFR method — with an MAE of 32.53m — when compared to the other feature subsets. Compared to the base model (model 10), this is an improvement of 8.01m. It is unexpected to see that this subset of features works best because it includes the two least important features according to the permutation importance: the slimness and number of vertices (see Table 4.7). The slimness, however, did express a negative correlation of 0.3 to the building height showing that there is some linear relationship present (see Figure 4.12a). Considering the MLR and SVR methods, both perform best with model 1 which only includes the area of the building footprints. They achieved MAEs of 32.18m and 31.81m respectively. The feature analysis showed that the area expresses a positive correlation to the building height of 0.4 (see Figure 4.12a). Both MLR and SVR are only capable of detecting linear relationships, providing a possible indication of why they perform well when trained on this feature subset. The MLR method improved 4.91m compared to the base model, while the SVR method improved 5.02m. The RFR method achieved the highest accuracy improvement of the three methods, but the ranking in terms of which machine learning method performs the best for CBD of Seattle is still the same: SVR performs best, followed by MLR and RFR.

The results for Seattle, using the prediction model that is trained on both area morphologies, is shown in Figure 5.20b. All three methods perform worst using model 2, which uses only the footprint area. This contrasts the observations for the CBD prediction model, where both the MLR and SVR methods performed best using this feature subset. The correlation matrix for the combined training dataset shows a low positive correlation of 0.11 (see Figure 4.19a), which is a possible reason why they underperform. However, also the RFR method struggles to detect the relationship between the footprint area and the building height, indicating that there is also no real non-linear relationship present. The violin plot in Figure 4.22 showed very similar kernel density plots per category for the building area, giving an explanation of why the different machine learning methods might struggle to find patterns in the data. For the RFR method, subset 7 performs best, which combines the number of adjacent buildings, area morphology and number of vertices. The first two features are the most important features for both the correlation- and permutation-based feature importance (see Table 4.9), while the number of vertices is regarded as quite insignificant in both methods. The method achieves an MAE of 32.16m. Compared to the base model (model 10), the RFR method realises an accuracy improvement of 8.15m. For the MLR method, subset 4 works best with an MAE of 32.45m, even though the scores for the multicollinearity among the features are very high. The method achieves an accuracy improvement of 0.39m compared to the base model. Lastly, for the SVR method the base model is the best choice, achieving an MAE of 34.87m.

So, using the single prediction network with the morphology as a (possible) feature achieves the best results using the RFR method, followed by the MLR and SVR methods. This contrasts the results of the split prediction network, where the SVR method performed best and the RFR method performed the worst.

The second test area is Portland, Oregon. The results for the different feature subsets — with a prediction model that is trained on only rural and suburban data — is shown in Figure 5.21a. The performance of the three machine learning methods does in general lie quite close to each other. Model 9, including all possible features, works best for the RFR method with an MAE of 1.42m. The MLR and SVR methods, however, do both perform best using model 4. Both methods have an MAE of 1.43m, which is an improvement of 0.24m and 0.2m compared to the base model (model 9) for MLR and SVR respectively. Model 4 includes the top five features based on the correlation values (see Table 4.7). This subset of features also expresses multicollinearity among the features, but to a lesser degree than for the CBD prediction network that was used on the Seattle dataset. Given these results, the RFR method with feature subset 9 is the best choice, but the difference with the results for the MLR and SVR methods using feature subset 4 is almost negligible.

The feature subset tests for the prediction model that is trained on both area morphologies shows more extremes in the MAEs (see Figure 5.21b). For models 8 and 9, both the RFR and MLR methods have extremely high MAEs compared to the other feature subset models. Model 8 only includes the slimness and number of vertices, and model 9 only the area. The correlation matrix in Figure 4.19a shows that the slimness, area, and number of vertices only express a very low linear relationship to the building height. This makes it difficult for the MLR method to create a 'good' model. However, the SVR method, which also tries to detect linear relationships, performs relatively well for these two models. For the RFR method, the slimness and number of vertices are deemed the two least important features, while the area is the third most important feature (see Figure 4.21). Despite that, it fails to accurately predict the building heights with only the footprint area. For the RFR method, as was the case for the rural and suburban prediction model, the model containing all features performs the best (model 10). An MAE of 1.42m is achieved, but using model 7 also provides a similar prediction accuracy. This model contains the number of adjacent buildings, combined with the slimness and number of vertices. This combination stands out because the model with only the latter two features was the worst-performing model for the RFR predictor. For the MLR method, model 3 performs the best with an MAE of 1.46m. Compared to the base model (model 10), this is an improvement of 0.31m. The SVR method, like the RFR method, has the best performance when all features are included and it results in an MAE of 1.41m. This makes the SVR method the best performing method of the three, followed by the RFR and MLR methods.

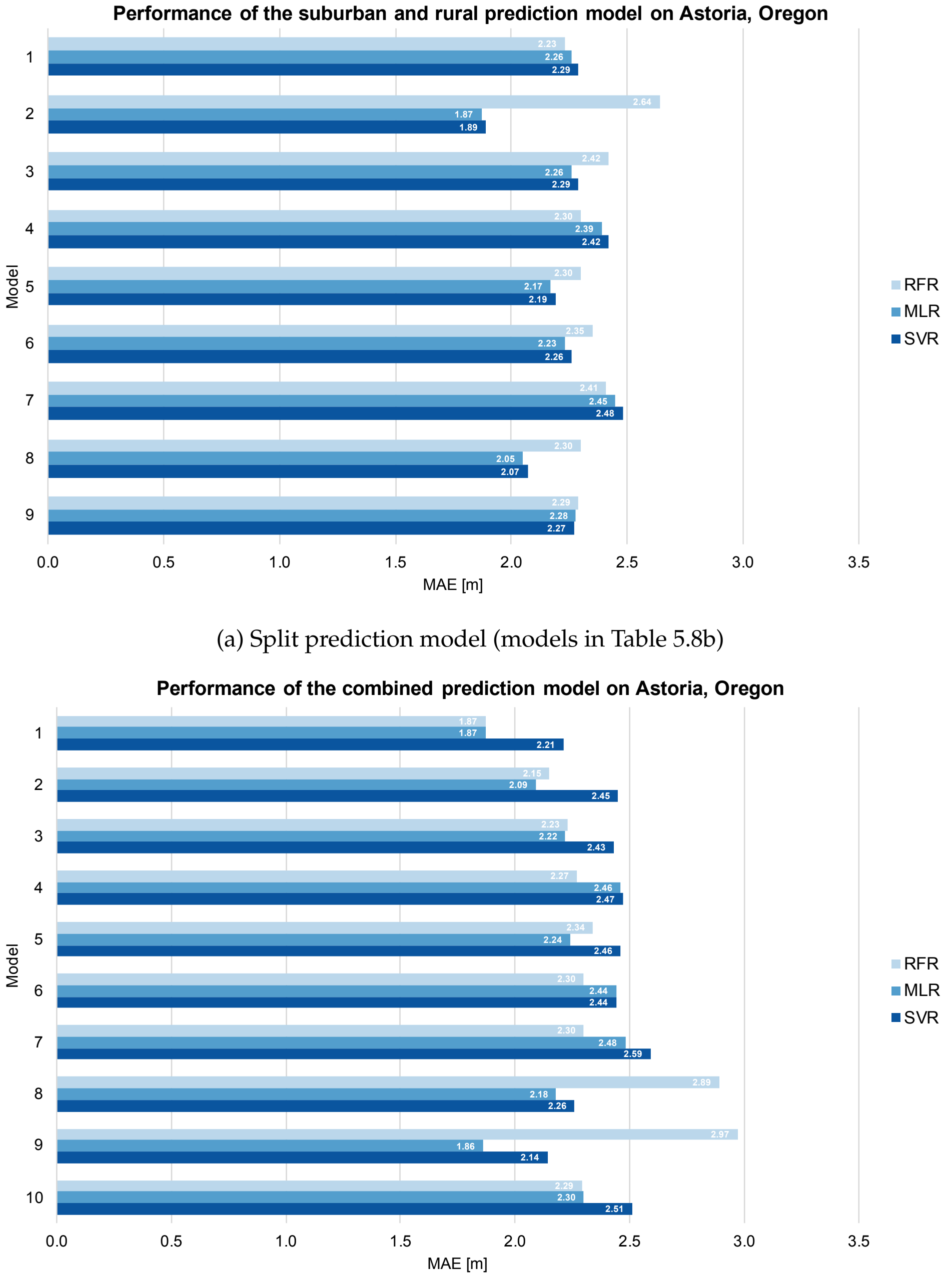


(a) Split prediction model (models in Table 5.8b)

(b) Combined prediction model (models in Table 5.9)

Figure 5.21: Results of the feature subsets for the three machine learning methods tested on Portland, Oregon.

The last test area is Astoria, Oregon. Figure 5.22a shows the results for the different prediction models that are trained on the rural and suburban data. For the RFR method, model 1 performs the best with an MAE of 2.23m. This subset only includes the number of adjacent buildings, and it provides an improvement of 0.06m compared to the base model (model 9). The MLR and SVR methods both show significantly lower errors than the RFR method. The best feature subset for these methods is captured by model 2, where only the footprint area is included. For the MLR method the MAE is 1.87m, while the MAE for the SVR method is 1.89m. Compared to the base model (model 9), this is an improvement of 0.41m and 0.38m for MLR and SVR respectively. Even though Pearson's correlation coefficient shows there is no linear relationship between the footprint area and building height (see Figure 4.12b), the two methods that depend on linear relationships perform the best. As was the case for Portland, the RFR method does not perform well based on only the footprint area.



(a) Split prediction model (models in Table 5.8b)

(b) Combined prediction model (models in Table 5.9)

Figure 5.22: Results of the feature subsets for the three machine learning methods tested on Astoria, Oregon.

Lastly, the combined prediction model, which is trained on both area morphologies, is analysed. Figure 5.22b shows that, in general, the MAEs are higher for the combined prediction model compared to a model that is split based on the area morphologies. However, for the RFR method the error can be greatly reduced. Model 1 — including only the area morphology — has an MAE of 1.87m. Compared to the base model (model 10), an improvement of 0.42m is obtained. For the MLR and SVR methods, the feature subset of model 9 works best with an MAE of 1.86m and 2.14m respectively. Compared to the base model, the accuracy of the MLR predictions improved by 0.44m and the SVR predictions by 0.37m. Model 9 is based on only the area and also proved to work best for both machine learning methods in case a split prediction network was used. The linear relationship between the footprint area and the building height is slightly more present in the combined training data, but it is still not significant (see Figure 4.19a). For the MLR method, model 1 also proves to work well, with an MAE of 1.87m; only 0.01m higher than the MAE of model 9.

This analysis for the three test areas highlights the challenges of optimising the selection of a subset of features that works best in all test cases. For the RFR method with the prediction model that is split based on the area morphology, different subsets worked best for the different test areas. For Seattle and Astoria, the MLR and SVR methods performed best when only the footprint area is included. For the Portland test area, however, these two machine learning methods benefited most from using the top five features based on Pearson's correlation coefficient.

For the combined prediction model, no such trends were found. Another thing that should be noted is that features that were deemed insignificant based on their correlation coefficient or permutation importance score, proved to be useful in reducing the MAE in several cases. The number of feature subsets tested here is limited and other subsets could lead to even better prediction models.

5.7 Influence of Height Percentiles

In this last analysis, the focus is put on the use of different height percentiles for training and testing the prediction models. For the three test areas, it is checked which height percentile leads to the best results. Three height percentiles are tested: the 50th, 75th and 90th. Table 5.10 shows for each test area how many buildings are present for the given height percentile that — according to the results from `3dfier` — have a building height greater than or equal to 3m. The lower the percentile, the fewer buildings that are present in the dataset because more buildings have extremely low (<1m) building heights and are thus excluded. The average height is also computed for the footprints. For Portland and Astoria, the difference between the 50th and 90th percentile lies around 1m. The building heights in Seattle, however, show a lot bigger difference of almost 15m between those same height percentiles. As opposed to Portland and Astoria, the Seattle dataset contains many tall buildings which could cause this difference.

		Height percentile		
		50th	*75th*	*90th*
Seattle	*# buildings*	216	219	223
	Avg. height [m]	33.10	41.98	47.86
Portland	*# buildings*	107,361	127,884	132,017
	Avg. height [m]	4.75	5.25	5.70
Astoria	*# buildings*	3,683	3,712	3,723
	Avg. height [m]	6.34	6.95	7.33

Table 5.10: The number of building footprints that are present for each test area for a certain height percentile. Also the average height of these building footprints is provided. Only buildings with a height greater than or equal to 3m are considered.

During the training stage, the height percentile is changed accordingly for the training datasets that have this information available. These datasets include Wilson, St. George, Cedar City, Junction City, Hood River, and Scio. For Toronto and New York City, it is not known which height percentile is used for the buildings that have their height derived from LiDAR. The heights for these two datasets are thus not changed during the testing process.

		CBD model			Combined model		
Regressor		*50th*	*75th*	*90th*	*50th*	*75th*	*90th*
RFR	*MAE [m]*	45.61	41.60	40.54	45.14	40.19	39.74
	MAPE [%]	279.73	242.10	224.93	272.91	230.72	216.87
	RMSE [m]	52.00	48.87	48.58	51.56	47.59	47.91
	RMSPE [%]	384.97	379.84	361.21	375.06	358.69	351.15
MLR	*MAE [m]*	42.27	38.09	37.09	21.71	27.89	32.84
	MAPE [%]	269.77	232.68	218.27	113.44	114.82	117.57
	RMSE [m]	47.10	44.73	44.73	35.28	43.31	49.66
	RMSPE [%]	368.29	352.52	341.09	177.74	184.41	186.50
SVR	*MAE [m]*	42.03	37.91	36.83	20.79	29.18	34.88
	MAPE [%]	267.70	230.67	216.12	66.61	74.67	78.68
	RMSE [m]	46.87	44.52	44.44	38.44	48.19	55.25
	RMSPE [%]	364.68	348.27	337.03	95.48	104.50	107.51

Table 5.11: Height prediction results when different height percentiles are used for training and computing the errors for Seattle, Washington. All nine geometric features are included in the prediction models. A distinction is made between the model that is only trained on CBD data and a model trained on all data with the area morphology as an extra feature.

The first test area we look at is Seattle, Washington. Table 5.11 shows the results for the two different prediction models: one that is trained with only the CBD data, and the other that is trained with all training data with the area morphology as an extra feature. All nine geometric features are included in these tests. Looking at the results of the different height percentiles for the CBD model shows that the 50th percentile performs worst, followed by the 75th and 90th percentiles. As was concluded in Section 5.2, the machine learning methods tend to overestimate the building heights in this area. Using the highest percentile thus lowers the prediction errors because of the higher average building height, making it the best fit for this case.

The combined prediction model shows some different results. For the RFR method, the 90th still works best, but for the MLR and SVR methods the 50th percentile does now provide the lowest MAEs. The linear machine learning methods can more accurately predict the lower building heights, as was shown in Figure 5.4. The difference in errors between the 50th and 75th percentile of the two linear methods and the RFR method is significant. Including all training data with the morphology as a feature seems to enable these two methods to better detect relations in the data.

The results for Portland, Oregon are shown in Table 5.12. Here, the split model is trained based on only the suburban and rural data because no CBD was detected. For both the split and the combined prediction model, the 50th percentile works best, followed by the 75th and 90th percentiles. The suburban and rural prediction model tends to predict the building heights well for suburban areas, but not for areas with taller buildings. Using the 50th percentile for training and testing data could significantly lower the height for areas with taller buildings as we saw with Seattle. Lower building heights in these regions could therefore also influence the MAE because the predictions and the ground truth lie closer to each other than for higher height percentiles.

The last test area is Astoria, Oregon. Table 5.13 shows the results for the different height percentiles. As for Portland, the split network is trained on the suburban and rural training data. Again, in all cases the 50th percentile works best, followed by the 75th and 90th percentiles. This could have a similar reason to the one described for Portland. For the MLR and SVR methods, the split prediction network performs slightly better than the combined prediction network. In Section 5.2, the results showed that for the taller buildings in the dataset the height is (almost) always underestimated. Again, with a lower height percentile the error for these building will be lower, and thus the MAE will decrease.

		Suburban / Rural model			Combined model		
Regressor		*50th*	*75th*	*90th*	*50th*	*75th*	*90th*
RFR	*MAE [m]*	1.34	1.35	1.42	1.35	1.36	1.42
	MAPE [%]	28.92	26.00	24.69	29.21	26.19	24.92
	RMSE [m]	2.05	2.12	2.36	2.06	2.12	2.36
	RMSPE [%]	36.79	33.86	32.56	37.20	34.02	32.76
MLR	*MAE [m]*	1.38	1.49	1.67	1.49	1.59	1.77
	MAPE [%]	28.48	26.85	27.27	31.27	29.24	29.30
	RMSE [m]	2.13	2.29	2.61	2.24	2.38	2.68
	RMSPE [%]	34.63	32.46	32.59	40.55	37.39	36.72
SVR	*MAE [m]*	1.36	1.46	1.65	1.28	1.30	1.41
	MAPE [%]	28.18	26.38	26.79	26.81	23.53	22.64
	RMSE [m]	2.09	2.26	2.58	1.99	2.09	2.39
	RMSPE [%]	34.05	31.75	31.91	30.85	27.55	26.64

Table 5.12: Height prediction results when different height percentiles are used for training and computing the errors for Portland, Oregon. All nine geometric features are included in the prediction models. A distinction is made between the model that is only trained on suburban and rural data and a model trained on all data with the area morphology as an extra feature.

		Suburban / Rural model			Combined model		
Regressor		*50th*	*75th*	*90th*	*50th*	*75th*	*90th*
RFR	*MAE [m]*	1.91	2.16	2.29	1.91	2.16	2.29
	MAPE [%]	27.77	28.57	28.92	27.74	28.48	28.91
	RMSE [m]	2.60	2.86	2.99	2.57	2.82	2.98
	RMSPE [%]	36.95	37.03	36.06	36.16	35.88	35.95
MLR	*MAE [m]*	1.86	2.11	2.28	1.92	2.15	2.30
	MAPE [%]	28.85	28.98	28.09	30.30	29.97	29.91
	RMSE [m]	2.42	2.72	2.65	2.51	2.76	2.94
	RMSPE [%]	35.48	34.94	33.78	39.04	36.96	36.32
SVR	*MAE [m]*	1.85	2.10	2.27	1.97	2.29	2.51
	MAPE [%]	28.42	28.68	29.08	27.43	29.07	30.13
	RMSE [m]	2.40	2.71	2.92	2.60	2.95	3.21
	RMSPE [%]	34.73	34.37	34.58	31.68	33.14	34.21

Table 5.13: Height prediction results when different height percentiles are used for training and computing the errors for Astoria, Oregon. All nine geometric features are included in the prediction models. A distinction is made between the model that is only trained on suburban and rural data and a model trained on all data with the area morphology as an extra feature.

These results for the three test areas show that for regions with lower buildings, the 50th height percentile seems to work best, while for CBDs generally the 90th percentile is the better fit. But, the question that remains is whether such conclusions can be drawn from analysing the MAEs of the prediction results. It is not known which of the three height percentiles lies closest to true building heights, because this information is not available. To check the quality of the height percentiles, it would be necessary to have another dataset available that has the confirmed heights for all buildings. In this way, the error between the height percentiles and the true building height can be computed and the 'correct' percentile can be selected. However, the process of selecting a building height percentile becomes obsolete if the true building heights are available.

6 Conclusions

In this chapter, the research questions are answered based on the findings in Chapter 5. Next, the contributions to the state-of-the-art are provided, followed by a discussion about the implementation of the methodology. The limitations highlighted in the discussion form the basis for possible future work.

6.1 Overview Research Questions

In this section the research questions that are defined in Section 1.1 are reviewed. Each question is analysed based on the evidence that is presented in the previous chapters.

1. *Can the 125 million USA building footprints be assigned a height without making use of height data, and what accuracy can be achieved?*

Yes, it is possible to predict the building height for all buildings in the conterminous USA in the absence of height data. Depending on the machine learning method and the type of prediction model, different accuracies are achieved (see Section 5.2). Among the few test areas, the best results are obtained for Portland, Oregon. The Support Vector Regression (SVR) method, using the combined prediction model (i.e. area morphology as additional feature), achieved a Mean Absolute Error (MAE) of 1.41m. For the Central Business Districts (CBDs), it proved to be challenging to accurately predict the building heights. The lowest MAE is 32.84m for the Multiple Linear Regression (MLR) method with the combined prediction model. Using a different configuration for the features can sometimes reduce these errors (see Section 5.6). For the Seattle CBD, for example, the split prediction model achieves an MAE of 31.81m with the SVR method when only the footprint area is used as a feature.

Also, all predictions for the conterminous USA are performed in under 6 minutes for the Random Forest Regression (RFR) method, under 26 seconds for the SVR method, and under 24 seconds for the MLR method (see Section 5.1). The combined prediction model is slower compared to the models that are split based on the area morphology.

1.1. *What methods can be used to assess the accuracy of the building height estimations? And when are the estimations deemed accurate enough?*

Reference models with ground truth building heights are required to assess the accuracy of the building height estimations. With this information, four different measures can be derived to assess the prediction results: 1) Mean Absolute Error (MAE), 2) Mean Absolute Percentage Error (MAPE), 3) Root Mean Square Error (RMSE), and 4) Root Mean Square Percentage Error (RMSPE) (see Section 3.5). Where the MAE and RMSE provide their errors in metres, the MAPE and RMSPE define the error as a percentage value. The latter two can provide better insight into the error relative to the actual building height. In terms of when the estimations are deemed accurate enough, there are no guidelines specifying anything for 3D city models. The CityGML specification contains a suggestion for LOD1 models to have an error of maximum 5m. Therefore, the accuracy results are compared to this recommendation.

1.2. *What relations are present between the different geometric properties of the building footprints and the building height? And which subset is deemed 'optimal' for predicting building heights?*

In terms of linear relationships between the features and the building height, the CBDs and the rural and suburban areas express different characteristics (see Section 4.4.1). For the CBDs the length and width of a building express a relatively high correlation to the building height, followed by the area and shape complexity. For the suburban and rural areas, the relationships are less clear; only the number of adjacent buildings expresses a moderate correlation to the building height. The permutation importance — derived from Random Forests (RFs) — can identify non-linear relationships. According to this measure, the building area and width are still very important for the CBDs, while for the suburban and rural regions the number of adjacent buildings stays the most important feature. But, also the area and number of neighbours are deemed more important for predicting the building height.

When all data is considered — with the area morphology as an additional feature — slightly different observations are found (see Section 4.4.2). As was the case in the suburban and rural areas, the linear relationships to the building height are not very strong. However, the area morphology does express a significant linear relationship to the building height. It is the most important feature in both the correlation analysis and permutation importance.

The analysis of the feature subsets in Section 5.6 showed that there is not one subset of features that works best for all prediction models and test areas. For the prediction models that are split up based on the area morphology, the MLR and SVR methods perform well for Seattle and Astoria when only the footprint area is included. For the RFR method, the best subset of features highly depends on the test region. In the prediction model that uses that area morphology as an extra feature, even fewer trends are present. The observations also showed that features that are of low importance can still positively impact the prediction results if they are used in certain combinations with other features. Further testing is required to see if there are other feature subsets that work better than the ones presented in this thesis. So, from my observations, I cannot pinpoint one subset of features that is 'optimal' for predicting the building heights.

Lastly, it must be noted that the use of only geometric features could introduce some problems, mainly for the MLR and SVR methods. Because all features are derived from the building footprints, certain features express strong relations to other features. The width and length of a building are, for example, highly correlated to the footprint area. The multicollinearity that is introduced can influence the performance of the prediction models because it becomes less clear how each of the features is separately associated with the target variable.

1.3. *Are the geometric properties of the building footprints as training features sufficient for meeting the accuracy requirements?*

To some extent. Whether the geometric properties are sufficient depends on the area morphology. For CBDs the errors are extremely high (i.e. more than 30m), while the suburban and rural areas express errors that are well within the 5m limit that is proposed in the CityGML specifications (see Section 5.2). Other factors, such as the data that is used for training the prediction models, can also influence these results. If the data does not cover a wide range of possible real-world scenarios, it can be more difficult for the machine learning methods to detect certain relations in the data. Further testing is required to check whether this is also the case for the training data used in this thesis project.

1.4. *What other features, besides the geometric properties of the building footprints, can be used in the machine learning algorithms to estimate building heights? And does including these features, even if they are incomplete, improve the accuracy of the estimations?*

The trial for the city of Denver showed that census data can be used to enrich the building footprints. These features include the average household size, average household income, and population density. The footprints dataset also contained information about the building type, which would be classified more as cadastral information. In addition, OpenStreetMap (OSM) provides information about the amenities in a certain area, and a Digital Elevation Model (DEM) can be used to extract an initial height estimation for each building footprint. The analysis in Section 5.5 showed that for the suburbs, only the raster building height estimate expressed a relatively high positive correlation to the true building height. In the CBD there are stronger linear relationships present. The raster building height and the number of amenities have a relatively high positive correlation to the true building height, while for the average household size and population density it is a negative correlation. When the CBD and suburb data are combined, the trends that are found are very similar to those in the suburbs.

Even though the relation to the building height is not always the clearest, including extra features does lead to accuracy improvements for all three machine learning methods. The RFR method achieves a sub-metre accuracy of 0.96m when the suburban prediction model is used. Scaling these extra features to the entire USA can pose serious difficulties, mainly because the data is scattered around over local governments or over many separate non-geospatial datasets.

1.5. *What methods can be used for scaling the machine learning techniques to the whole of the USA?*

The scaling problem consists of two parts. First, the runtime of the actual training and predicting stages is reduced by running processes in parallel on several CPUs (see Section 3.2.2). Additionally, the hyper-parameters of the RFR and SVR methods are tuned which can also improve the efficiency of the models (see Section 4.3).

The second aspect considers the different types of prediction models. Section 3.2.1 showed that CBDs express different characteristics for the building heights than the more rural and suburban regions. A distinction is therefore made between these two area morphologies. In this thesis, two approaches for the prediction models are implemented. The first creates separate prediction models, one for each area morphology. The second establishes one prediction model and uses the area morphology as an extra feature in this process. The results showed that no single approach works best, but that it depends on the area that is analysed (see Table 5.3).

In addition to these tests, it is also reviewed how a combined prediction model performs when the area morphology is excluded. Again, the results are mixed. With this prediction model, the RFR method for Seattle performs slightly better than the two models described above. The same is the case for the MLR and SVR methods for Astoria, Oregon. Even though the results vary, I believe that making the distinction between the two area morphologies is an important step to take. Further improvements should be implemented to increase the predictive power of the prediction models, especially for the CBDs (see Section 6.4).

6.2 Contributions

This thesis project builds upon popular machine learning techniques and applies them to the building height inference problem. The concepts that are presented in the methodology are not necessarily novel, but I believe they contribute to the state-of-the-art in the following ways:

- **Scalability analysis**: Two approaches for incorporating different area morphologies into the machine learning problem are proposed. The results showed that scaling based on the area morphologies is not a straightforward problem. The proposed methodology forms a basis which can be extended to further optimise the results (see Section 6.4). Also, this research proved that the building height inference problem can be applied to large datasets containing millions of building footprints and that the whole prediction pipeline can be performed within a reasonable time.

- **Exploit geometric features**: In comparison to the work presented by Biljecki et al. [2017], this thesis includes five additional geometric features that are derived from the 2D building footprints (see Section 3.4.1). All features are analysed in terms of their relation to the building height. Furthermore, different combinations of features are tested and their influence on the prediction results is analysed. The analysis showed that there is not one feature subset that works best for all prediction models and test regions. Selecting a suitable feature subset for the entire USA can thus be challenging.

- **Influence of roof reference points**: The height predictions for the three test areas are analysed based on the use of the 50th, 75th, and 90th height percentiles (see Section 5.7). This showed that for the CBDs, the 90th percentile did in general produce the lowest MAE. For the more rural and suburban areas, the 50th percentile is the better fit. These results are influenced by the predictive capabilities of the prediction models, because the CBD prediction model tends to overestimate the building heights, while the rural and suburban model tends to underestimate them.

6.3 Discussion

The results of the implementation of this thesis project are mainly positive. It is possible to predict the height for all buildings in the conterminous USA within a reasonable time. But, for the CBDs there is plenty of room for improvement because of the high deviations from the ground truth. These high errors limit the usefulness of the LOD1 models for other applications; even for visualisation purposes, the models will not provide an accurate representation of the real world. For the suburban and rural areas, the models are also fairly crude and the lowest achieved MAE is 1.41m. For applications that do not require high-quality data, these models can still be useful. One example is the visualisation application stated before, which includes navigational purposes. The buildings from the 3D city model can be used to provide a simplified representation of the surroundings, making it easier for a user to orientate themselves.

Another consideration is that the methodology of this thesis still partly relies on building height information in the training stage. For existing datasets, it is often unknown what the accuracy of the building heights is. This is also the case for the Toronto and New York City datasets that are used in the training process in this research. Furthermore, the training and reference models that are derived from LiDAR data also introduce a certain degree of uncertainty. The building height computation is affected by several factors such as adjacent buildings. Also, the footprint and LiDAR data might not perfectly match. As described in Section 4.1, the `USBuildingFootprints` dataset aggregates buildings into bigger blocks in the case of many adjacent buildings. Therefore, the height for these buildings is determined based on data that actually belong to several buildings. This can introduce errors in both the training and reference data. The last consideration for the reference models is the different height percentiles that can be used to compute the building height. Using the 90th percentile will include higher data points, resulting in taller buildings than when, for example, the 75th percentile is used. As a consequence, the behaviour of the prediction model is influenced by this decision and therefore affects the building height predictions (see Section 5.7).

The methodology proposed in this thesis is, in theory, general enough to be applied outside the USA. A pre-requisite is that the study area has 2D footprints available; OSM, for example, offers open data for many parts of the world. Also, training data that contains building heights is required. If this requirement cannot be met, data from similar areas elsewhere could be used to train the prediction model(s). When taking this approach, a careful study of the target areas should be conducted to make sure they express similarities in the sense of their urban layout and building heights. The detection process of the different area morphologies is currently specific to the USA. The methodology could be applied in other places if both a Digital Elevation Model (DEM) and a neighbourhood dataset are available for the selected region. The use of a specific neighbourhood dataset is a limitation of this research. In the USA specifically, these neighbourhoods are often large and combine several area morphologies. This results in true CBDs not being detected because other buildings in its neighbourhood are significantly lower. The use of a grid-based approach could potentially solve this problem, see Section 6.4. It could also be considered to incorporate more types of area morphologies, instead of only the distinction between the CBDs and the rural and suburban areas. For areas outside the USA, it is also possible that there is no need to discriminate between different area morphologies. Not all countries have cities with many high-rise buildings. Therefore, separate decisions should be made for each use-case.

The selected training data and features (subsets) are two other crucial aspects to consider because they influence the building height estimations. The feature analysis and feature subset selection are performed based on the trends that are found in the training data. Features in other datasets could potentially express very different relationships to the building height, which would lead to differently trained prediction models. The differences between the characteristics of the CBDs and the rural and suburban data already highlight this point (see Section 4.4.1). As a result, the subset of features that is important in a training dataset is not necessarily relevant for other places in the USA. The data selection process is thus of high importance, and a wide variety of urban and/or rural regions must be included.

The last limitation of this research is that the non-geometric features have only been tested for a small test area. However, the trial for the city of Denver did show that including these kinds of features could significantly improve the prediction results (see Section 5.5). The use of these features should, therefore, be expanded to the entire USA. In general, the RFR method did benefit most from adding the new features, potentially because of its ability to detect non-linear relations in the data. For future implementations, it is therefore recommended to focus on the RFR method, even though it is not the fastest method of the three.

A final remark is about the comparison of the three machine learning models to the Open City Model (OCM) (see Section 5.4). It is like comparing apples to oranges. Whereas I estimate the building height for all buildings in the USA, the OCM, combines data from various sources. They only perform height predictions in the areas where no building height information is available. It is unknown how accurate the building heights of these other data sources are. For the cities of Portland and Seattle, no comparison could be performed because the OCM did not provide any height predictions for these areas. For Astoria, however, they did use machine learning. The prediction models presented in this thesis perform slightly better than the OCM for this area. But, this small test region is not representative of the entire USA. In the future, a comparison like this should be avoided because of the different nature of both models.

6.4 Future Work

Based on the limitations discussed in Section 6.3, several recommendations to improve the methodology are defined. The suggestions are the following:

- **Area morphology detection**: The detection process should be improved. The neighbourhoods used in this research are relatively large (see Figure 4.8) because they correspond to the neighbourhoods that are present in the real world. These neighbourhoods can, therefore, contain different area morphologies. This introduces problems with the fixed thresholds that are used to extract the CBDs; either too many or too little CBDs are detected. A next step could be to research whether a different neighbourhood division improves the detection results of the proposed workflow in Section 4.2.3. These new regions can be manually defined around the areas of interest (i.e. areas with high elevation values), or a grid-based approach can be used. Some examples of possible grid layouts are shown in Figure 6.1. Statistics could be derived for each grid cell, and these are then used to perform the area morphology classification. This approach is extendable to all regions in the USA, and even outside it if proper elevation data is available.

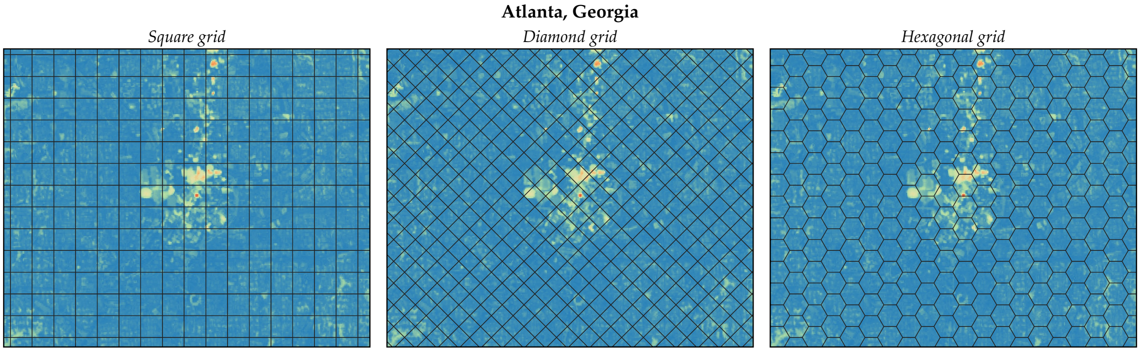


Figure 6.1: Grid-based approach for detecting area morphologies (500x500m cells) for Atlanta, Georgia. Blue indicates lower heights, red higher heights.

Also, the current area division is very black and white; only a distinction is made between the CBDs and the remaining regions. A more suitable division could be to use at least three classes: CBDs, suburbs, and rural areas. The latter two classes are then not aggregated into one group any more. A possible tool for this classification, in addition to the DEMs, could be the use of road network data. Meijer et al. [2018] harmonised several datasets into one global roads dataset. It includes five hierarchical road types, including highways, primary roads, secondary roads, tertiary roads, and local roads. Cities and suburbs are often characterised by a higher density in road networks than the more rural regions. This fact could possibly be exploited to extract the different area morphologies.

- **Additional non-geometric features**: The non-geometric features should be incorporated for the entire USA. Proprietary spatial `ArcGIS` data[1] contains census information for all states in the USA. Information on the building types could be extracted from open data sources like OSM. The latter requires a way of transferring the properties from the OSM buildings to the `USBuildingFootprints` dataset. Both a centroid or area overlap method could be used for this purpose. However, there is always the risk that a feature is assigned to the wrong building footprint, especially if the two footprint datasets deviate a lot from each other.

  In addition to the census data and the building types, it could be interesting to incorporate the shadows in satellite imagery as an indicator for the building height [Shao et al., 2011; Liasis and Stavrou, 2016]. Shadows with a longer length to width ratio could indicate taller buildings. This method requires knowledge about both the date and time the image was taken, but also about the altitude of the sun at that moment in time. In urban areas, difficulties with extracting the correct shadow per building may arise because of overlapping shadows.

  Lastly, a recent study by Li et al. [2020] mapped the 3D building structure for Europe, the USA, and China using RF models. Building footprints, heights, and volumes are estimated on a 1km^2 resolution. They use various sorts of satellite data in combination with other datasets such as road networks. The use of this data could potentially also be useful for the topic investigated in this thesis. Also, the height estimations of the 1km^2 grid could provide an initial indication of the building heights as an extra feature in the training process. However, it must be kept in mind that the data has a coarse resolution.

- **Feature subset selection**: Currently, subsets of features are selected based on an analysis of their linear relationship to the building height and their importance in the RF. The results in Section 5.6 showed that features that express low importance can still have a positive influence on the building height predictions. Therefore, more feature subsets should be tested. An exhaustive search algorithm can be used to test all possible feature combinations. It is important to consider, however, that the results can be biased towards the test areas that are used to examine the height estimation errors.

  In addition to the subsets of the geometric features, it is also beneficial to test subsets of features for the non-geometric features that are proposed in this thesis (see Section 3.4.2). There likely exists a subset of features that performs better than including all features in the prediction model.

- **Training data**: For the training data, an even greater diversity of possible building areas should be added. This is particularly true for the CBDs because the prediction errors for these areas are high. A possible explanation is that the training dataset is not versatile enough and that it does not cover the different scenarios it has to predict the height values for. However, constructing a high-quality training dataset is a difficult process.

- **Testing data**: Similar to the training data, the testing data should be increased to cover a broader range of test scenarios. Particular focus should be put on the CBDs, where attention must be paid to the city layout and the diversity in building heights. The suburban areas have been covered relatively well by the Portland dataset, but the rural test dataset is also still relatively small.

  Also, it could be insightful to test the influence of more granular building footprints. The dataset that is currently used omits certain building features that would usually be present for the individual buildings. The geometric features that are extracted for the buildings will be different, influencing the prediction results.

[1] See examples here: `https://www.arcgis.com/home/search.html?q=2019%20USA&showFilters=true&focus=layers`

A Reproducibility Self-Assessment

A.1 Marks for the Criteria

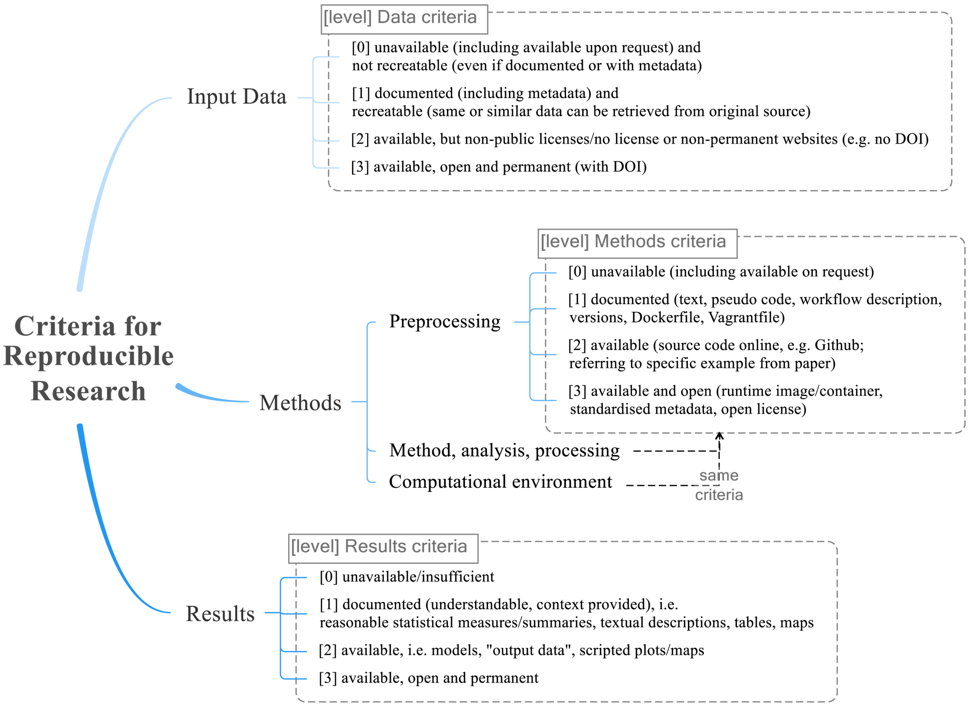

Criteria for Reproducible Research
Input Data
[level] Data criteria
[0] unavailable (including available upon request) and not recreatable (even if documented or with metadata)
[1] documented (including metadata) and recreatable (same or similar data can be retrieved from original source)
[2] available, but non-public licenses/no license or non-permanent websites (e.g. no DOI)
[3] available, open and permanent (with DOI)
Methods
Preprocessing
[level] Methods criteria
[0] unavailable (including available on request)
[1] documented (text, pseudo code, workflow description, versions, Dockerfile, Vagrantfile)
[2] available (source code online, e.g. Github; referring to specific example from paper)
[3] available and open (runtime image/container, standardised metadata, open license)
Method, analysis, processing
Computational environment
same criteria
Results
[level] Results criteria
[0] unavailable/insufficient
[1] documented (understandable, context provided), i.e. reasonable statistical measures/summaries, textual descriptions, tables, maps
[2] available, i.e. models, "output data", scripted plots/maps
[3] available, open and permanent

Figure A.1: Reproducibility criteria as specified in [Nüst et al., 2018].

Based on the five reproducibility criteria specified in [Nüst et al., 2018] (see Figure A.1), the evaluation of each criterion for the work presented in this thesis is shown in Table A.1. Because often more than one score applied, several scores are indicated and a more in-depth analysis of the scores is provided in Section A.2.

	Criterion	Score
1.	Input data	2 - 3
2.	Preprocessing	1 - 2
3.	Method, analysis, processing	1 - 2
4.	Computational environment	1 - 2
5.	Results	1 - 2

Table A.1: Evaluation of the five criteria specified in Figure A.1.

A.2 Self-Reflection

In terms of the *input data* that is used in this research, all data is openly available. However, not all datasets are provided with a Digital Object Identifier (DOI), making it possible that some of the data may change or is unavailable in the future. The data does require *preprocessing* before it can be used for the purposes of this thesis. A textual description of this process is provided (see Chapter 4), while the automated parts are made available online via GitHub[1]. An example of a non-automated process included combining the building heights generated with `3dfier` with the 2D building footprints, while the automated preprocessing includes, for example, generating the unique identifiers. For the *method*, *analysis*, and *processing* part, only open source software tools are used and proprietary software is avoided. The source code for the implementation of the methodology and most of the analysis is available in the GitHub repository. The documentation on this web page also specifies the software libraries used (i.e. *computational environment*), including the versions to improve the reproducibility of this research. Lastly, in terms of the *results* of this thesis project, the maps presented in Chapter 5 are created based on a manual process using `QGIS`. Most other results in this chapter are automatically generated using the scripts provided on the GitHub page. The output data, however, is not provided simply because of the large amounts of data. Users are free to request these files or generate the output files themselves based on the processes stated in this thesis.

All in all, independent researchers should be able to reproduce this research, but some difficulties can arise with the manual steps that are present. An advantage in terms of reproducibility is that the project does solely rely on open data sources and that the source code is available online.

[1] Source code available on GitHub: `https://github.com/ImkeLansky/USA-BuildingHeightInference`

B Dataset Information

This appendix contains extra information on the datasets used in this research. Table B.1 contains information on the LiDAR datasets that are used for creating the reference models, including links to the data download page and the metadata. Table B.2 shows the three datasets for Denver, Colorado that contain the non-geometric features based on census data. Table B.3 contains the OCM datasets that are used for the three testing areas. Lastly, Table B.4 contains information about the city limits boundary data.

Table B.1: The LiDAR datasets used for creating the reference models, including their data source.

Nickname	Dataset name	Survey date	pts/m^2	Source
Wyoming 1	Snake River, Wyoming	2008	4.80	(1)
Wyoming 2	Teton Conservation District, Wyoming Lidar	2008	2.62	(2)
Utah 1	2015-2017 State of Utah Lidar Acquisition	2015-2017	7.17	(3)
Utah 2	Utah Geological Survey Lidar	2011	1.91	(4)
Washington 1	PSLC King County 2016-2017 LiDAR	2016-2017	8.00	(5a,b)
DOGAMI	Oregon Department of Geology and Mineral Industries Lidar Program Data	2007-2010	10.30	(6)

(1) Data & metadata: https://doi.org/10.5069/G9W95737
(2) Data & metadata: https://doi.org/10.5069/G9F769GN
(3) Data & metadata: https://doi.org/10.5069/G9RV0KSQ
(4) Data & metadata: https://doi.org/10.5069/G90C4SPQ
(5a) Data: https://lidarportal.dnr.wa.gov/#47.59849:-122.32933:14
(5b) Metadata: https://lidarportal.dnr.wa.gov/download?ids=835

(6) Data & metadata: https://doi.org/10.5069/G9QC01D1

Table B.2: The Denver census datasets used for testing the non-geometric features.

Dataset name	Used attribute	Source
American Community Survey Tracts (2014-2018)	AVG_HH_INCOME	(1a,b)
Census Tracts (2010)	Avg_HH_Size	(2a,b)
Population Density (Census Tracts)	Population Density	(3a,b)

(1a) Data: https://www.denvergov.org/opendata/dataset/american-community-survey-tracts-2014-2018
(1b) Metadata: https://denvergov.org/media/gis/DataCatalog/american_community_survey_tracts_2014_2018/metadata/american_community_survey_tracts_2014_2018.xml

(2a) Data: https://www.denvergov.org/opendata/dataset/city-and-county-of-denver-census-tracts-2010
(2b) Metadata: https://www.denvergov.org/media/gis/DataCatalog/census_tracts_2010/metadata/census_tracts_2010.xml

(3a) Data: https://data-cdphe.opendata.arcgis.com/datasets/2128d5e4260a47c28b3fd124f79008a1_0

(3b) Metadata: https://www.cohealthmaps.dphe.state.co.us/arcgis/rest/services/CHE_MAPS/CDPHE_CHE_MAPS/MapServer/0

Table B.3: The OCM datasets used in the comparison of the height predictions of the machine learning methods and the predictions in the OCM.

City / Town	County name	Dataset name(s)	Source
Seattle*	King County	`Washington-53033-004`	(1)
		`Washington-53033-010`	(2)
		`Washington-53033-016`	(3)
Portland	Multanomah County	`Oregon-41051-000`	(4)
		`Oregon-41051-001`	(5)
		`Oregon-41051-002`	(6)
		`Oregon-41051-003`	(7)
		`Oregon-41051-004`	(8)
		`Oregon-41051-005`	(9)
		`Oregon-41051-006`	(10)
		`Oregon-41051-007`	(11)
Astoria	Clatsop County	`Oregon-41007-000`	(12)

* The CityGML and CityJSON files did not cover the same area, therefore separate datasets are provided

(1) CityJSON (2) CityGML
(3) CityJSON (4) CityGML, CityJSON
(5) CityGML, CityJSON (6) CityGML, CityJSON
(7) CityGML, CityJSON (8) CityGML, CityJSON
(9) CityGML, CityJSON (10) CityGML, CityJSON
(11) CityGML, CityJSON (12) CityGML, CityJSON

Table B.4: The datasets for city boundaries used for extracting the corresponding building footprints.

Dataset name	Cities	Source
`City Limits (all of Oregon)`	Portland, Astoria, Scio, Hood River, Junction City	(1a,b)
`Wambachers-OSM`	St. George, Wilson, Cedar City	(2)

(1a) Data: http://data-drcmetro.opendata.arcgis.com/datasets/1ab75ec6b10c40cf83c7b45449b02943_0?selectedAttributes%5B%5D=acres&chartType=bar&geometry=-124.423%2C45.367%2C-122.863%2C45.656
(1b) Metadata: `http://gis.oregon.gov/DAS/EISPD/GEO/docs/metadata/citylim_2011.xml`

(2) `https://wambachers-osm.website/boundaries/`

C Probability Density of the Building Heights for all Features

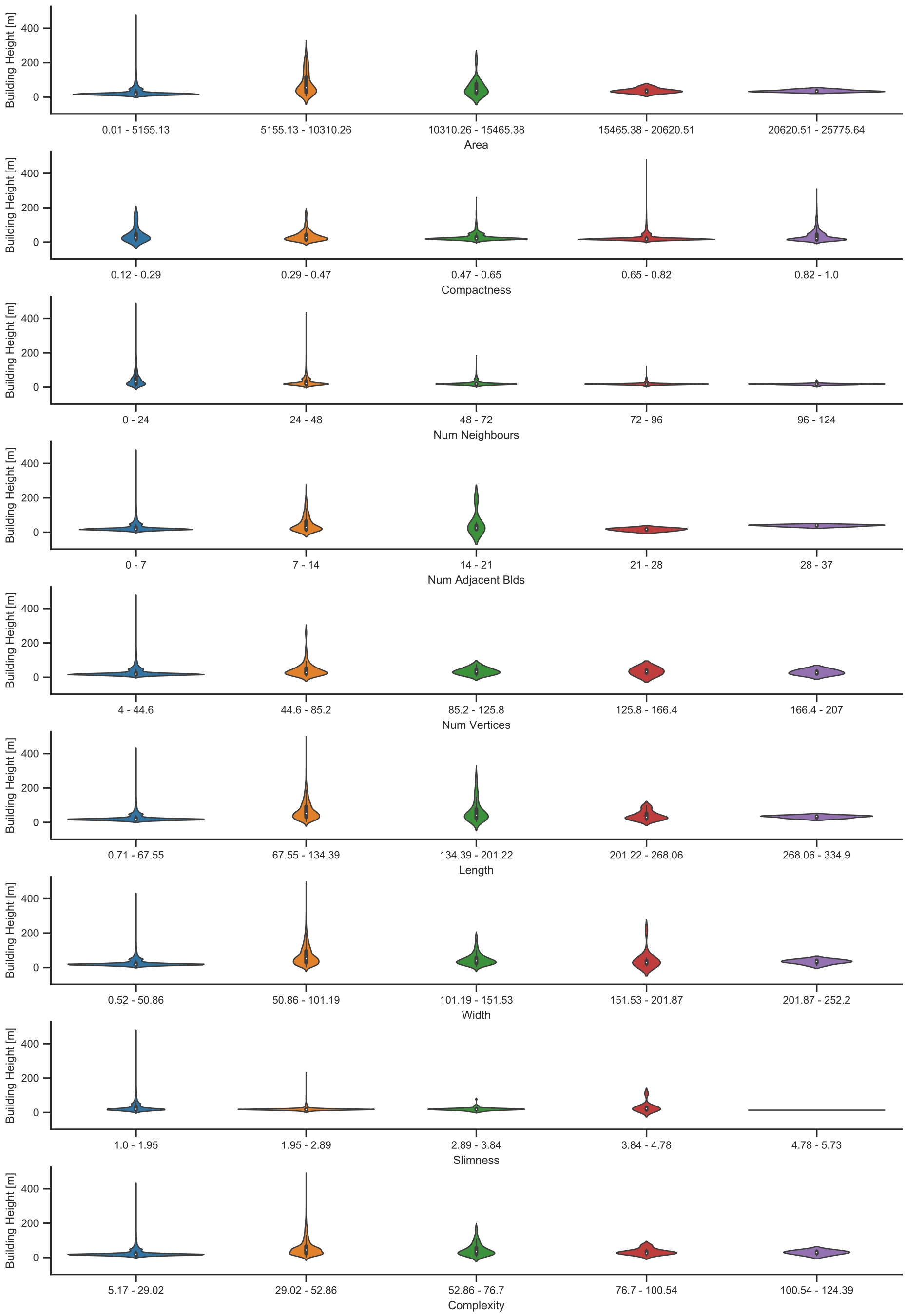


Figure C.1: Violin plot showing the probability density of the building height for each of the features in the CBD setting.

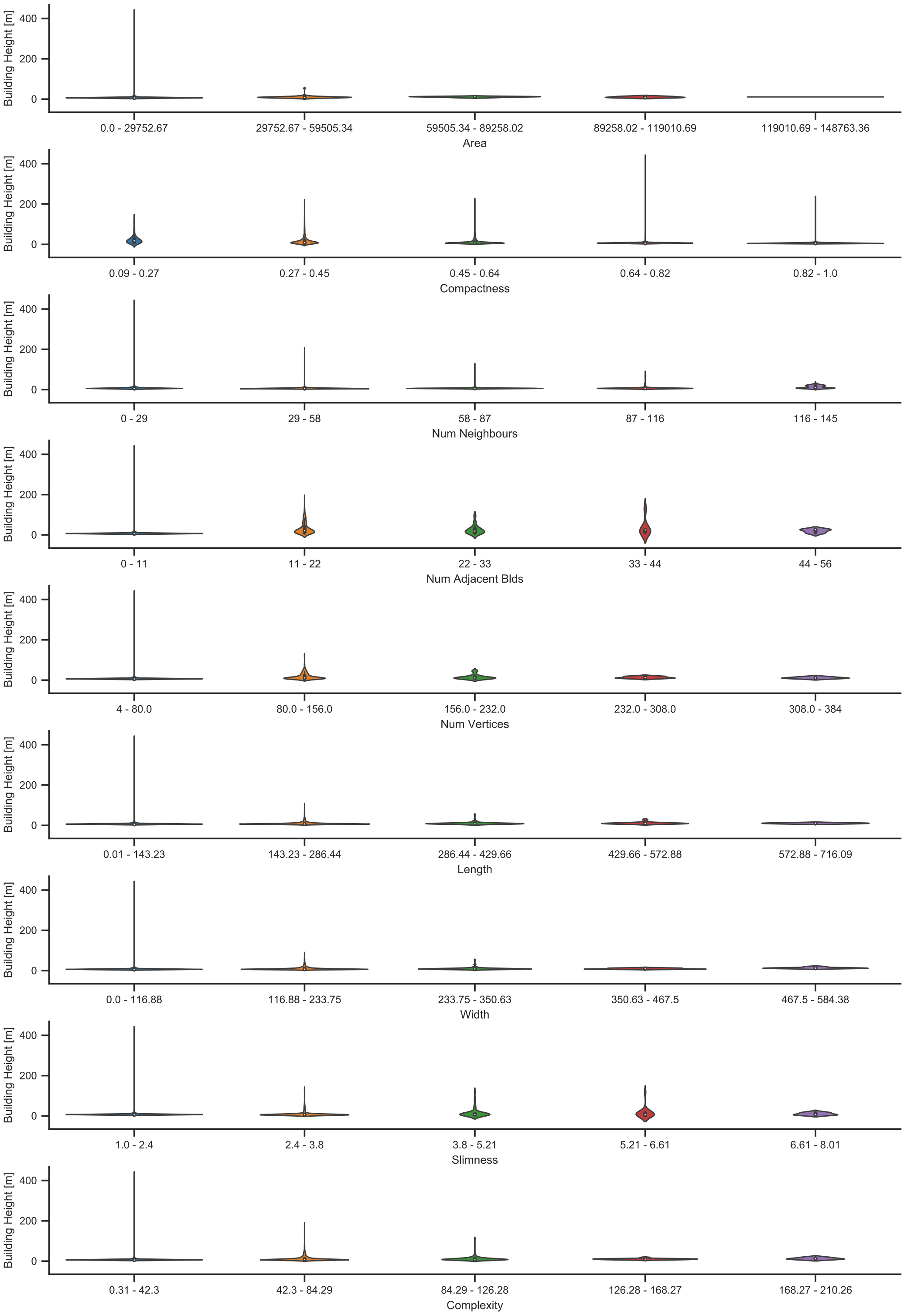


Figure C.2: Violin plot showing the probability density of the building height for each of the features in the suburban and rural setting.

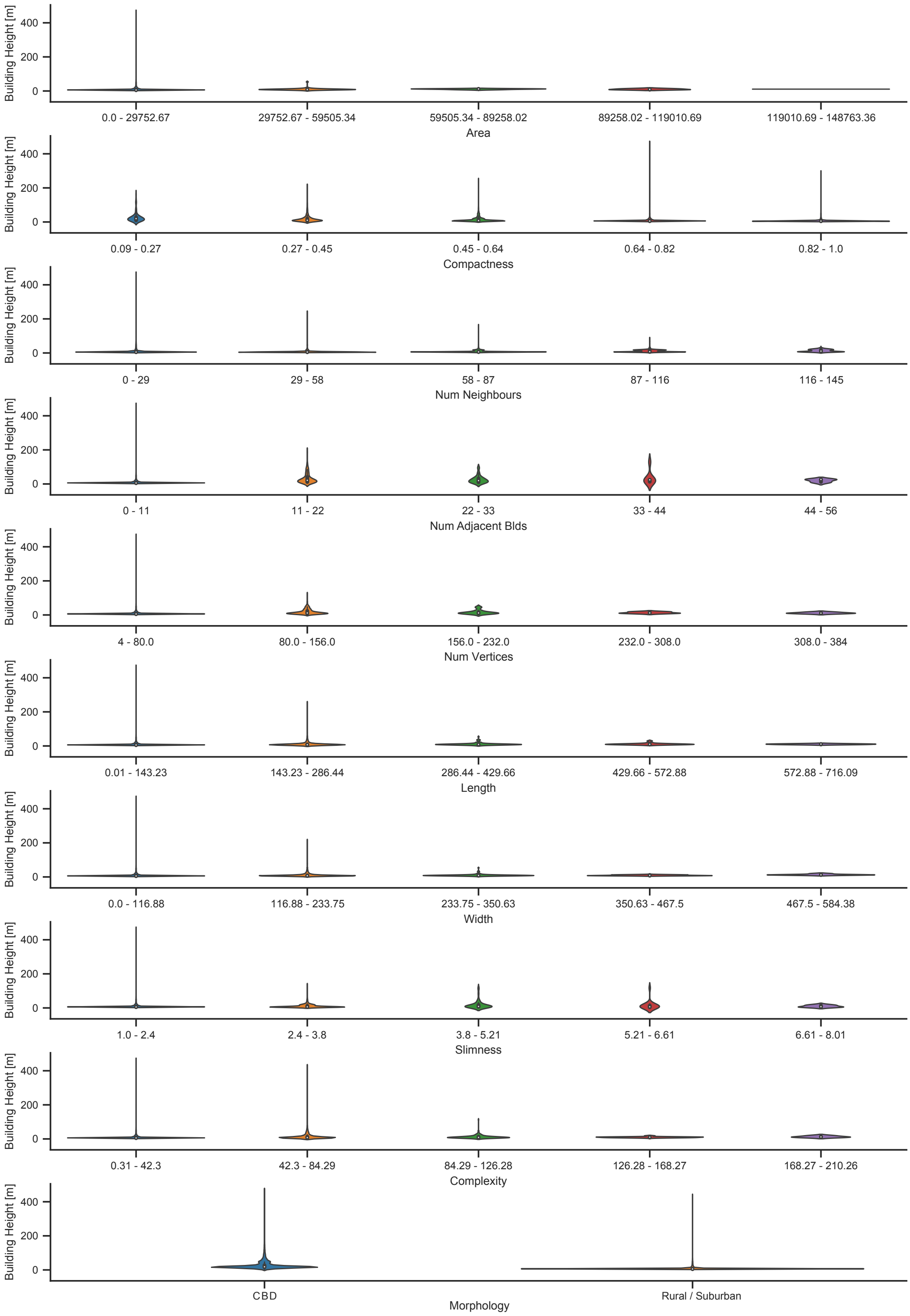


Figure C.3: Violin plot showing the probability density of the building height for each of the features in combined area morphology setting.

D Additional Results

D.1 Portland

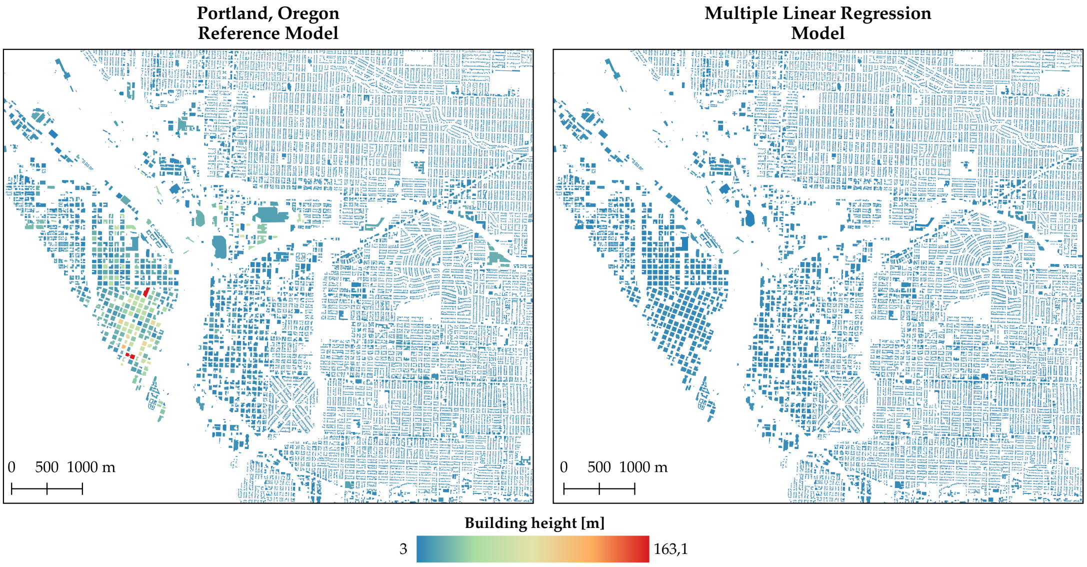


Figure D.1: Maps showing a comparison between the building heights in the reference model and the MLR building height predictions for the area around the city centre of Portland, Oregon. The MLR prediction model is trained on only the rural and suburban data.

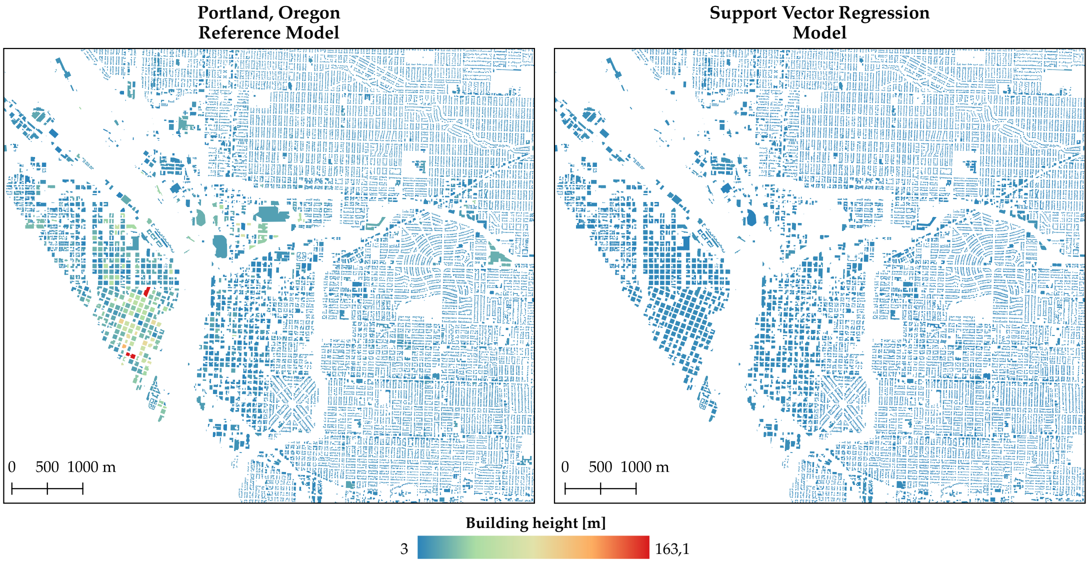


Figure D.2: Maps showing a comparison between the building heights in the reference model and the SVR building height predictions for the area around the city centre of Portland, Oregon. The SVR prediction model is trained on only the rural and suburban data.

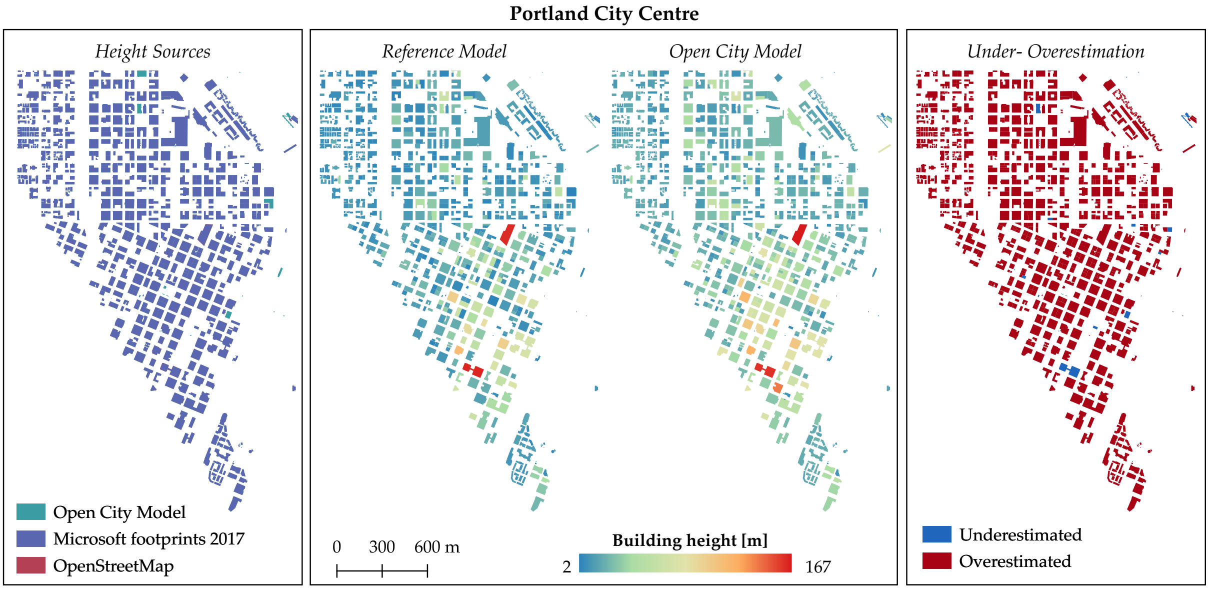


Figure D.3: Maps showing a comparison between the building heights in the reference model and in the OCM for the city centre of Portland, Oregon. Many of the height values are not from the OCM.

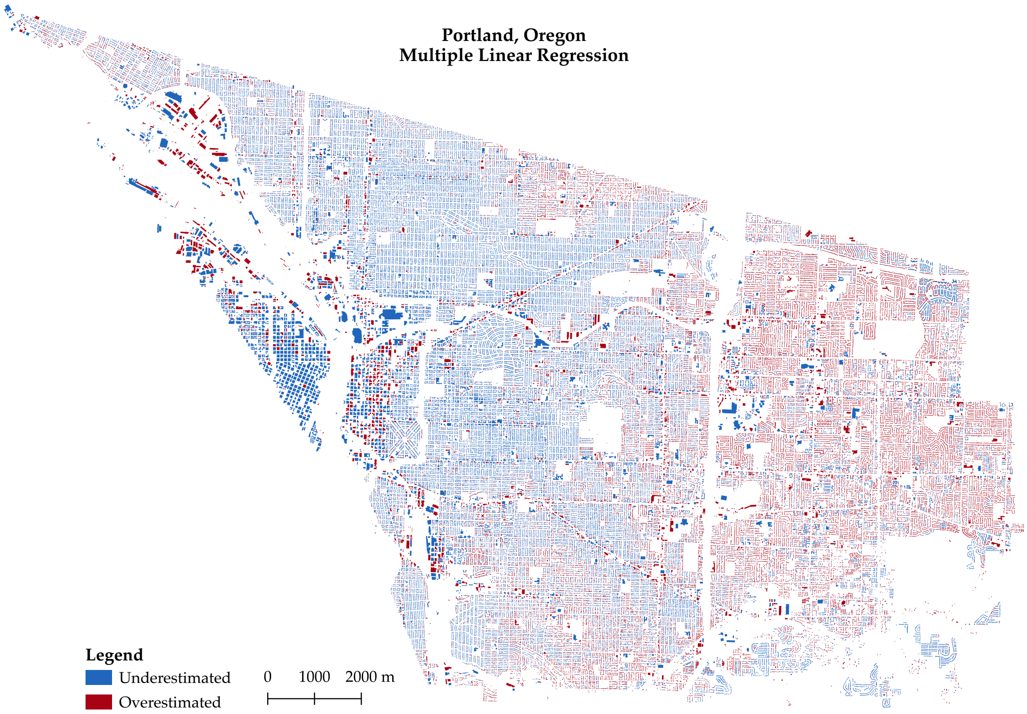


Figure D.4: Map showing the under- and overestimations of the MLR building height predictions for Portland, Oregon. The MLR prediction model is trained on only the rural and suburban data.

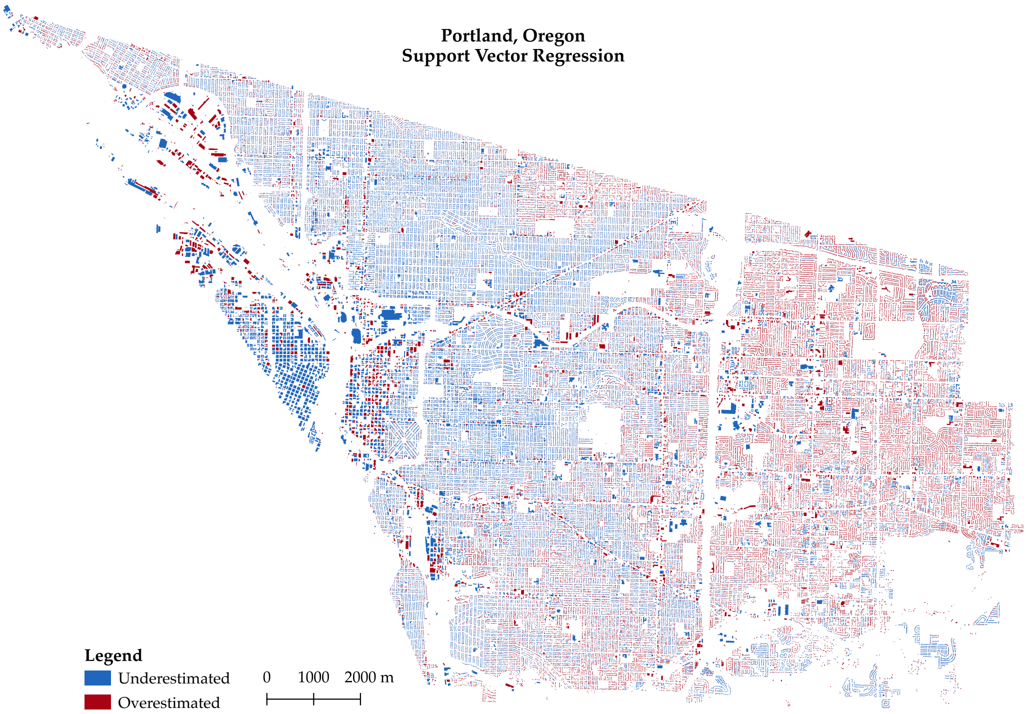


Figure D.5: Map showing the under- and overestimations of the SVR building height predictions for Portland, Oregon. The SVR prediction model is trained on only the rural and suburban data.

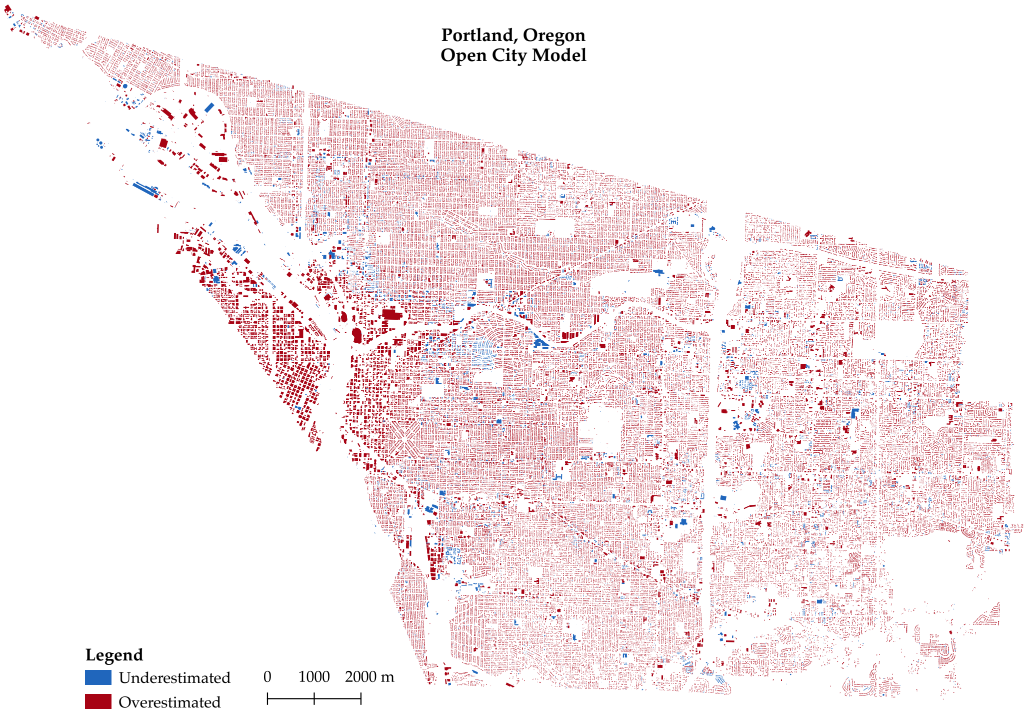


Figure D.6: Map showing the under- and overestimations of the OCM building heights for Portland, Oregon. The prediction model is trained on only the rural and suburban data.

D.2 Astoria

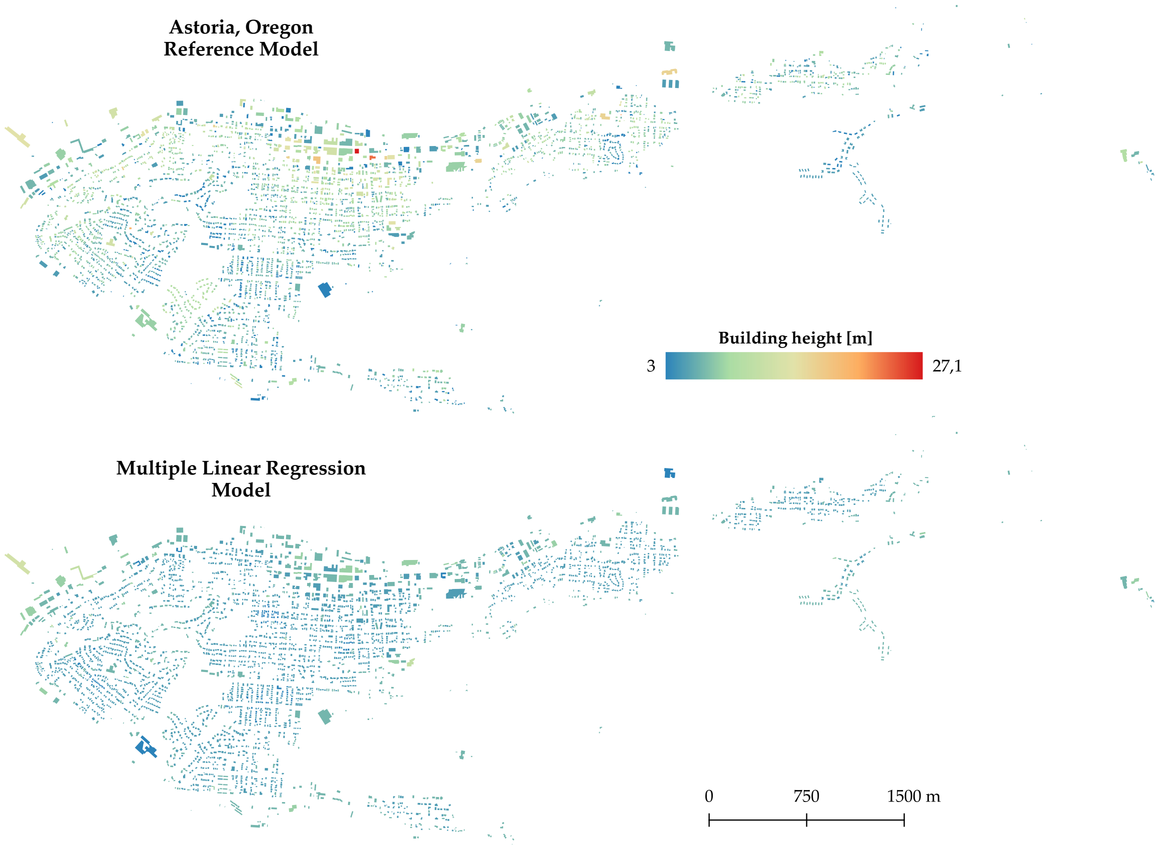


Figure D.7: Maps showing a comparison between the building heights in the reference model and the MLR building height predictions for the town of Astoria, Oregon. The MLR prediction model is trained on only the rural and suburban data.

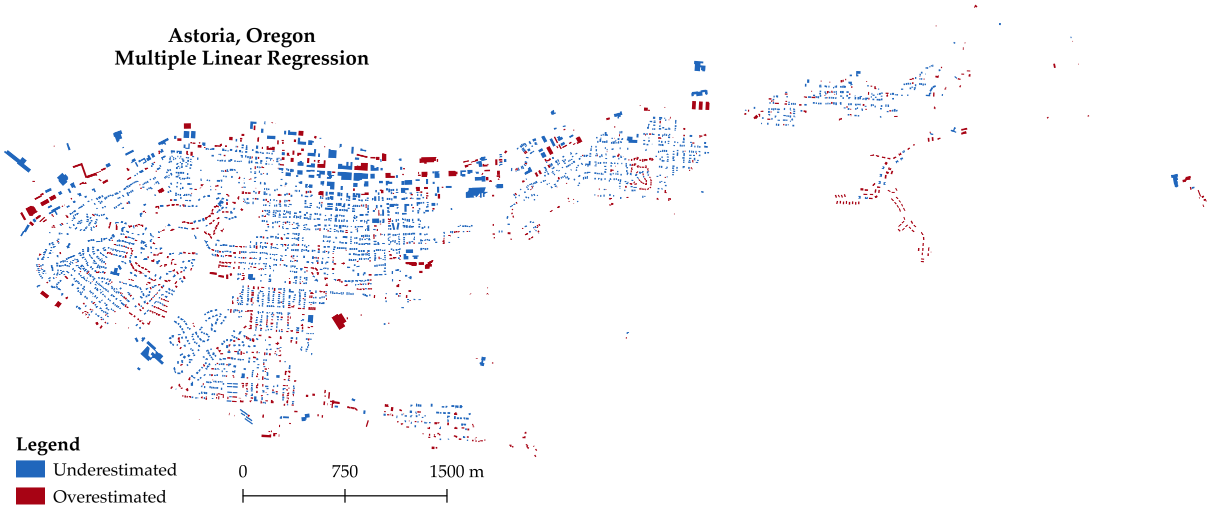


Figure D.8: Map showing the under- and overestimations of the MLR building height predictions for the town of Astoria, Oregon. The MLR prediction model is trained on only the rural and suburban data.

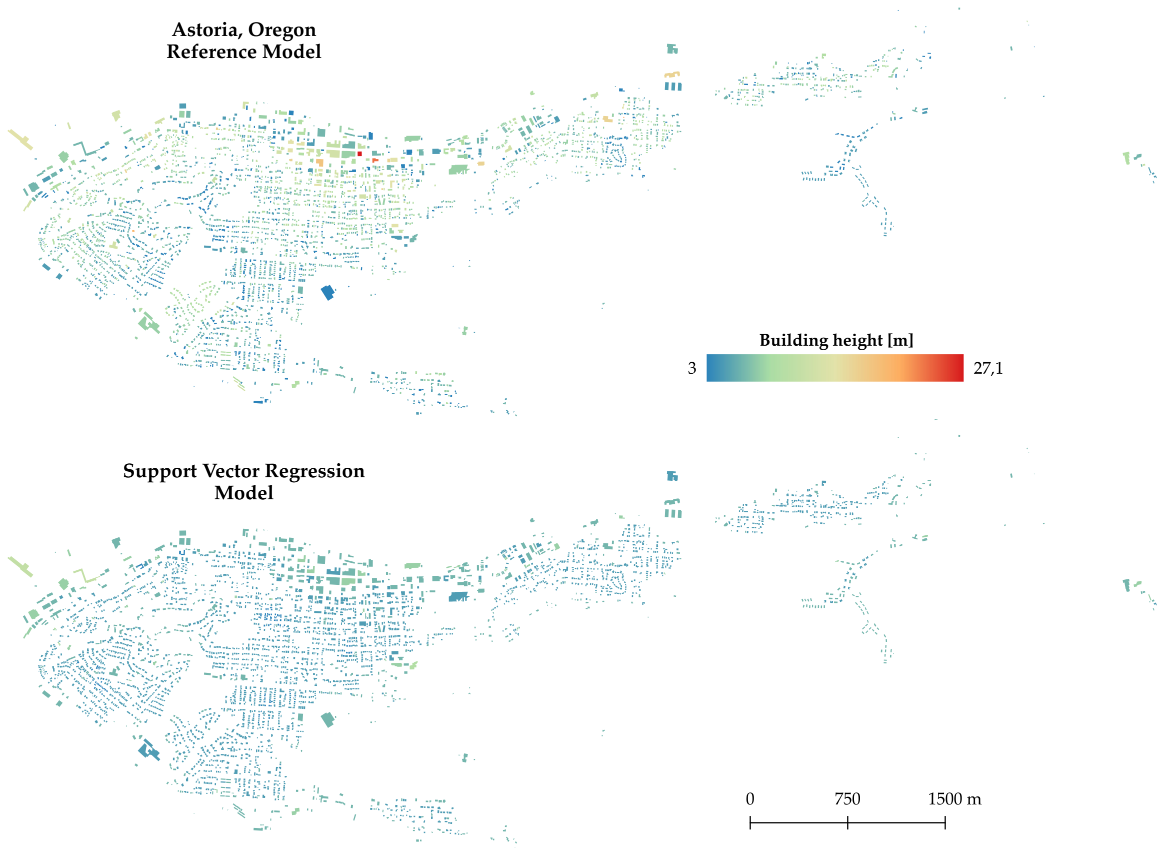


Figure D.9: Maps showing a comparison between the building heights in the reference model and the SVR building height predictions for the town of Astoria, Oregon. The prediction SVR model is trained on only the rural and suburban data.

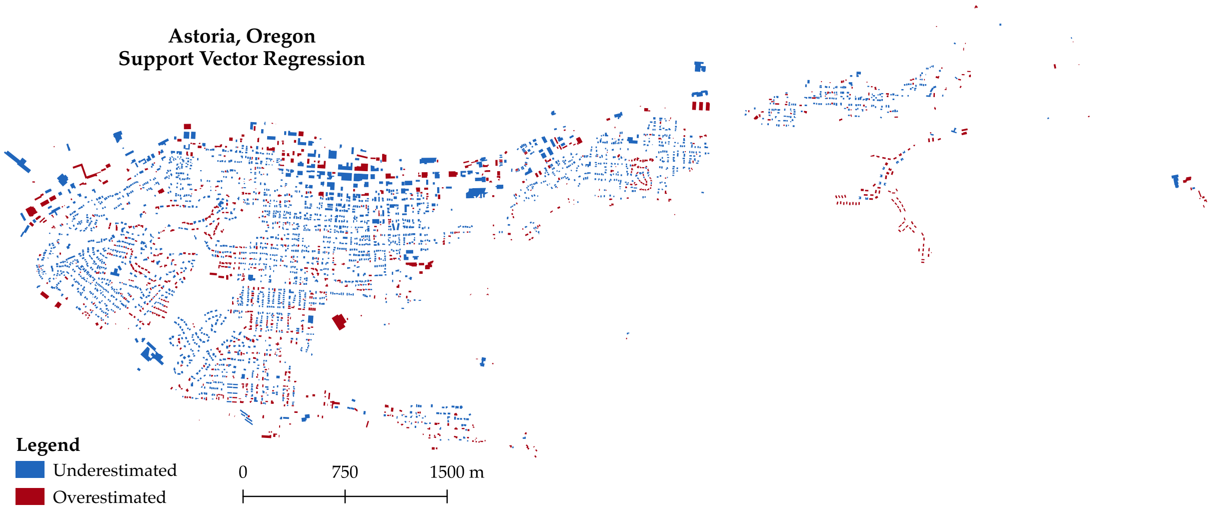


Figure D.10: Map showing the under- and overestimations of the SVR building height predictions for the town of Astoria, Oregon. The SVR prediction model is trained on only the rural and suburban data.

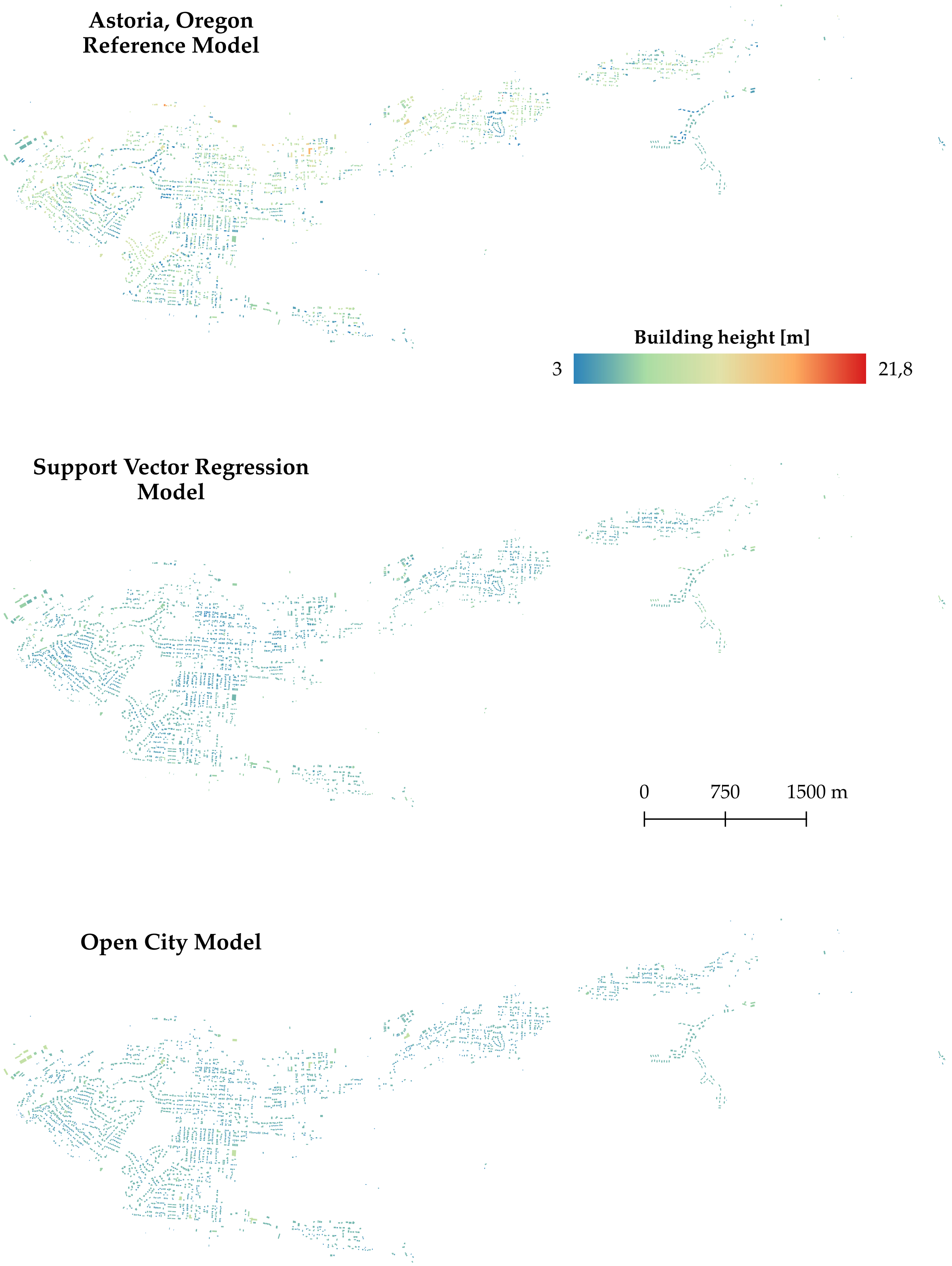


Figure D.11: Maps showing the difference between the building height predictions of RFR, MLR, and the OCM for Astoria, Oregon. The RFR and MLR prediction models are trained on only the rural and suburban data.

D.3 Feature Subsets

		Model #									
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Seattle	*MAE [m]*	33.62	42.29	35.13	37.99	38.55	35.81	32.16	40.00	35.56	39.74
	MAPE [%]	115.51	71.95	185.55	203.06	206.98	180.53	145.66	70.83	73.72	216.87
	RMSE [m]	51.02	63.02	46.96	47.57	46.67	48.39	45.73	60.75	56.05	47.91
	RMSPE [%]	184.88	76.05	313.39	337.28	333.57	336.30	245.71	76.53	95.81	351.15
Portland	*MAE [m]*	1.71	1.48	1.45	1.48	1.49	1.57	1.42	3.28	2.62	1.42
	MAPE [%]	34.82	27.00	25.70	26.31	26.37	28.39	24.67	66.25	53.31	24.92
	RMSE [m]	2.51	2.38	2.37	2.39	2.44	2.53	2.37	4.91	4.73	2.36
	RMSPE [%]	43.92	33.15	30.92	35.09	35.20	36.85	30.30	100.88	102.01	32.76
Astoria	*MAE [m]*	1.87	2.15	2.23	2.27	2.34	2.30	2.30	2.89	2.97	2.29
	MAPE [%]	25.58	26.89	27.48	28.59	29.60	28.86	28.21	44.21	44.78	28.91
	RMSE [m]	2.44	2.83	2.93	2.96	3.06	2.99	3.01	4.07	5.03	2.98
	RMSPE [%]	31.93	31.79	31.92	35.77	37.82	34.62	33.03	68.31	92.80	35.95

Table D.1: Results of the RFR method for different feature subsets with a combined prediction model. The subsets are defined in Table 5.9.

		Model #									
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Seattle	*MAE [m]*	33.62	42.15	34.10	32.45	33.00	33.94	33.84	40.05	38.14	32.84
	MAPE [%]	115.51	71.62	99.60	104.46	112.54	102.66	99.45	68.85	63.47	117.57
	RMSE [m]	51.02	62.91	52.87	50.10	50.41	52.34	52.49	61.11	59.05	49.66
	RMSPE [%]	184.88	75.75	152.35	161.24	176.91	158.95	152.59	75.07	69.19	186.50
Portland	*MAE [m]*	1.71	1.51	1.46	1.48	1.76	1.50	1.47	2.40	2.36	1.77
	MAPE [%]	34.82	28.29	25.91	24.50	29.52	25.58	24.24	50.52	50.43	29.30
	RMSE [m]	2.51	2.39	2.37	2.36	2.60	2.47	2.44	3.28	3.04	2.68
	RMSPE [%]	43.92	35.24	31.29	30.04	35.97	31.21	28.61	67.87	62.71	36.72
Astoria	*MAE [m]*	1.87	2.09	2.22	2.46	2.24	2.44	2.48	2.18	1.86	2.30
	MAPE [%]	25.58	26.46	27.37	30.04	30.00	30.03	29.93	33.50	29.65	29.91
	RMSE [m]	2.44	2.75	2.91	3.14	2.81	3.17	3.19	2.86	2.32	2.94
	RMSPE [%]	31.93	31.73	31.89	34.79	36.61	35.16	31.14	47.91	40.38	36.32

Table D.2: Results of the MLR method for different feature subsets with a combined prediction model. The subsets are defined in Table 5.9.

		Model #									
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Seattle	*MAE [m]*	35.12	42.78	35.79	35.07	35.09	35.76	35.64	41.77	41.38	34.88
	MAPE [%]	83.61	73.22	78.14	78.38	78.76	78.32	77.80	70.80	69.69	78.68
	RMSE [m]	55.23	63.39	56.34	55.44	55.51	56.29	56.13	62.57	62.06	55.25
	RMSPE [%]	117.95	77.31	105.07	105.94	106.87	105.51	104.63	75.21	73.82	107.51
Portland	*MAE [m]*	1.45	1.44	1.44	1.44	1.44	1.44	1.47	1.56	1.46	1.41
	MAPE [%]	25.90	23.81	23.90	23.60	23.60	23.80	23.44	28.70	26.61	22.64
	RMSE [m]	2.36	2.43	2.41	2.39	2.39	2.42	2.48	2.48	2.32	2.39
	RMSPE [%]	31.05	28.13	28.18	27.68	27.70	28.02	27.49	36.89	32.19	26.64
Astoria	*MAE [m]*	2.21	2.45	2.43	2.47	2.46	2.44	2.59	2.26	2.14	2.51
	MAPE [%]	27.20	29.51	29.25	29.70	29.67	29.37	31.01	28.69	26.61	30.13
	RMSE [m]	2.90	3.17	3.14	3.16	3.15	3.15	3.31	2.93	2.80	3.21
	RMSPE [%]	31.53	33.71	33.41	33.82	33.84	33.53	35.10	34.31	31.07	34.21

Table D.3: Results of the SVR method for different feature subsets with a combined prediction model. The subsets are defined in Table 5.9.

		Model #									
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Seattle	*MAE [m]*	38.83	38.49	40.42	39.47	40.02	37.29	41.14	38.12	32.53	40.54
	MAPE [%]	203.07	207.80	226.90	221.17	223.98	207.87	228.15	198.71	153.91	224.93
	RMSE [m]	47.90	46.51	48.35	46.73	47.43	46.11	49.30	46.24	46.48	48.58
	RMSPE [%]	312.09	326.43	381.90	349.55	352.58	328.14	358.85	322.73	285.54	361.21
Portland	*MAE [m]*	1.45	2.11	1.53	1.43	1.46	1.51	1.51	2.13	1.42	-
	MAPE [%]	25.70	42.25	26.54	25.17	26.19	26.31	26.06	41.84	24.69	-
	RMSE [m]	2.37	3.76	2.47	2.37	2.40	2.41	2.45	3.33	2.36	-
	RMSPE [%]	30.92	82.72	35.53	33.51	34.90	34.47	34.44	62.60	32.56	-
Astoria	*MAE [m]*	2.23	2.64	2.42	2.30	2.30	2.35	2.41 7	2.30	2.29	-
	MAPE [%]	27.48	38.16	30.41	28.78	29.02	29.60	30.11	31.95	28.92	-
	RMSE [m]	2.93	3.84	3.10	2.97	3.00	3.00	3.09	3.13	2.99	-
	RMSPE [%]	31.92	69.68	36.98	35.26	36.88	35.69	35.68	45.11	36.06	-

Table D.4: Results of the RFR method for different feature subsets with a split prediction model. The subsets are defined in Table 5.8.

		Model #									
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Seattle	*MAE [m]*	32.18	35.37	35.88	36.38	36.44	34.99	35.64	34.68	32.96	37.09
	MAPE [%]	155.48	201.80	206.07	213.37	214.84	203.27	203.72	194.54	133.25	218.27
	RMSE [m]	42.49	43.50	43.63	44.37	4.06	42.88	43.73	43.07	48.09	44.73
	RMSPE [%]	236.92	313.54	322.42	339.96	338.05	320.60	316.03	302.06	223.01	341.09
Portland	*MAE [m]*	1.45	1.68	1.44	1.43	1.58	1.46	1.46	1.58	1.67	-
	MAPE [%]	25.47	34.06	25.32	23.90	26.26	26.00	24.25	30.51	27.27	-
	RMSE [m]	2.38	2.46	2.36	2.36	2.47	2.36	2.43	2.42	2.61	-
	RMSPE [%]	30.66	42.98	30.44	28.33	31.39	31.70	28.69	38.34	32.59	-
Astoria	*MAE [m]*	2.26	1.87	2.26	2.39	2.17	2.23	2.45	2.05	2.28	-
	MAPE [%]	27.73	25.51	27.72	28.96	28.02	27.55	29.58	26.55	28.09	-
	RMSE [m]	2.96	2.44	2.95	3.08	2.78	2.91	3.16	2.66	2.65	-
	RMSPE [%]	32.21	31.65	32.16	33.19	33.43	32.17	33.81	31.77	33.78	-

Table D.5: Results of the MLR method for different feature subsets with a split prediction model. The subsets are defined in Table 5.8.

		Model #									
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Seattle	*MAE [m]*	31.81	35.13	35.71	36.02	36.27	34.79	35.36	34.45	42.96	36.83
	MAPE [%]	153.49	200.73	205.38	210.69	213.92	202.22	202.49	193.38	132.46	216.12
	RMSE [m]	42.20	43.27	43.48	43.85	43.81	42.72	43.47	42.86	48.16	44.44
	RMSPE [%]	234.32	312.76	321.69	333.22	335.97	319.82	315.03	301.05	221.44	337.03
Portland	*MAE [m]*	1.45	1.65	1.44	1.43	1.58	1.45	1.46	1.56	1.65	-
	MAPE [%]	25.13	33.25	24.97	23.67	26.12	25.63	24.04	29.92	26.79	-
	RMSE [m]	2.39	2.43	2.36	2.37	2.48	2.37	2.44	2.41	2.58	-
	RMSPE [%]	30.14	41.88	29.92	28.01	31.15	31.13	28.40	37.44	31.91	-
Astoria	*MAE [m]*	2.29	1.89	2.29	2.42	2.19	2.26	2.48	2.07	2.20	-
	MAPE [%]	28.03	25.48	28.02	29.30	28.11	27.81	29.96	26.64	29.08	-
	RMSE [m]	2.99	2.46	2.99	3.12	2.81	2.95	3.20	2.69	2.92	-
	RMSPE [%]	32.45	31.40	32.40	33.51	33.39	32.37	34.18	31.73	34.58	-

Table D.6: Results of the SVR method for different feature subsets with a split prediction model. The subsets are defined in Table 5.8.

Bibliography

Alamdari, A. R. S. A. (2006). *Variable Selection using Correlation and Single Variable Classifier Methods: Applications*, pages 343–358. Springer, Berlin, Heidelberg.

Alin, A. (2010). Multicollinearity. *WIREs Computational Statistics*, 2(3):370–374.

Angel, S., Parent, J., and Civco, D. L. (2010). Ten compactness properties of circles: measuring shape in geography. *The Canadian Geographer / Le Géographe Canadien*, 54(4):441–461.

Anh, P., Vu, C. T., Hung, B. Q., Thanh, N. T. N., and Ha, N. V. (2018). Preliminary Result of 3D City Modelling For Hanoi, Vietnam. In *NAFOSTED Conference on Information and Computer Science (NICS)*, pages 294–299.

Archuleta, C.-A. M., Constance, E. W., Arundel, S. T., Lowe, A. J., Mantey, K. S., and Phillips, L. A. (2017). *The National Map seamless digital elevation model specifications*, chapter 9. Techniques and Methods. U.S. Geological Survey.

Biljecki, F. and Dehbi, Y. (2019). Raise the Roof: Towards Generating LoD2 Models Without Aerial Surveys using Machine Learning. In *3D Geoinfo 2019 Proceedings*.

Biljecki, F., Ledoux, H., and Stoter, J. (2014). Height references of CityGML LOD1 buildings and their influence on applications. *Proceedings. 9th ISPRS 3DGeoInfo Conference*.

Biljecki, F., Ledoux, H., and Stoter, J. (2016a). An improved LOD specification for 3D building models. *Computers, Environment and Urban Systems*, 59:25–37.

Biljecki, F., Ledoux, H., and Stoter, J. (2016b). Generation of multi-LOD 3D city models in CityGML with the procedural modelling engine Random3Dcity. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-4/W1:51–59.

Biljecki, F., Ledoux, H., and Stoter, J. (2017). Generating 3D city models without elevation data. *Computers, Environment and Urban Systems*, 64:1–18.

Biljecki, F. and Sindram, M. (2017). Estimating Building Age with 3D GIS. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 4, pages 17–24.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., and Çöltekin, A. (2015). Applications of 3D City Models: State of the Art Review. *ISPRS International Journal of Geo-Information*, 4:2842–2889.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Information science and statistics. Springer, New York, NY.

Bousquet, O., von Luxburg, U., and Rätsch, G. (2004). *Advanced Lectures on Machine Learning*. Springer, Heidelberg, Germany.

Breiman, L. (1996). Bagging Predictors. *Machine Learning*, 24:123–140.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32.

Brovelli, M. A., Cannata, M., and Longoni, U. M. (2004). LIDAR Data Filtering and DTM Interpolation Within GRASS. *Transactions in GIS*, 8(2):155–174.

Brown, R. H., Barram, D. J., Ehrlich, E. M., and Scarr, H. A. (1994). *Census Tracts and Block Numbering Areas*, chapter 10.

BuildZero (2019). Open CityGML data for the United States. https://github.com/opencitymodel/opencitymodel (accessed: 16.05.2020).

Bureau, U. C. (2020). The Importance of the American Community Survey and the 2020 Census. https://www.census.gov/programs-surveys/acs/about/acs-and-census.html (accessed: 05.05.2020).

Chai, T. and Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3):1247–1250.

Chandrashekar, G. and Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28.

City Planning Toronto (2019). 3D Massing. https://open.toronto.ca/dataset/3d-massing/ (accessed: 15.12.2019).

Claesen, M. and de Moor, B. (2015). Hyperparameter Search in Machine Learning. *CoRR*, abs/1502.02127.

Coalition of Geospatial Organizations (2018). *Second Report Card on the U. S. National Spatial Data Infrastructure*. COGO.

Cohen, D. (2015). Understanding Population Density. https://www.census.gov/newsroom/blogs/random-samplings/2015/03/understanding-population-density.html (accessed: 05.05.2020).

Comber, A., Umezaki, M., Zhou, R., Ding, Y., Li, Y., Fu, H., Jiang, H., and Tewkesbury, A. (2012). Using shadows in high-resolution imagery to determine building height. *Remote Sensing Letters*, 3(7):551–556.

Dare, P. M. (2005). Shadow Analysis in High-Resolution Satellite Imagery of Urban Areas. *Photogrammetric Engineering and Remote Sensing*, 71(2):169–177.

Denver Regional Council of Governments (2019). Building Outlines (2016). https://www.denvergov.org/opendata/dataset/city-and-county-of-denver-building-outlines-2016 (accessed: 10.05.2020).

Douglas, D. H. and Peucker, T. K. (1973). Algorithms for the reduction of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122.

Duan, K., Keerthi, S., and Poo, A. N. (2003). Evaluation of simple performance measures for tuning SVM hyperparameters. *Neurocomputing*, 51:41–59.

Duch, W. (2006). *Filter Methods*, pages 89–117. Springer, Berlin, Heidelberg.

Eeftens, M., Beekhuizen, J., Beelen, R., Wang, M., Vermeulen, R., Brunekreef, B., Huss, A., and Hoek, G. (2013). Quantifying urban street configuration for improvements in air pollution models. *Atmospheric Environment*, 72:1352–2310.

ESRI (2014). The American Community Survey. https://www.esri.com/library/whitepapers/pdfs/the-american-community-survey.pdf.

Goetz, M. and Zipf, A. (2012). OpenStreetMap in 3D – Detailed Insights on the Current Situation in Germany. *Proceedings of the AGILE'2012 International Conference on Geographic Information Science*, pages 288–292.

Gregorio, F. D. and Varrazzo, D. (2001–2020). Psycopg – PostgreSQL database adapter for Python. https://www.psycopg.org/docs/ (accessed: 12.05.2020).

Gregorutti, B., Michel, B., and Saint-Pierre, P. (2017). Correlation and variable importance in random forests. *Statistics and Computing*, 27:659–678.

Gröger, G., Kolbe, T. H., Nagel, C., and Häfele, K.-H. (2012). *OGC City Geography Markup Language (CityGML) Encoding Standard*. Open Geospatial Consortium.

Grömping, U. (2009). Variable Importance Assessment in Regression: Linear Regression versus Random Forest. *The American Statistician*, 63(4):308–319.

Guney, C., Girginkaya, S. A., Cagdas, G., and Yavuz, S. (2012). Tailoring a geomodel for analyzing an urban skyline. *Landscape and Urban Planning*, 105(1):160–173.

Gunn, S. R. (1998). Support Vector Machines for Classification and Regression. Technical report, University of Southamptop.

Guyon, I. and Elisseeff, A. (2003). An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3:1157–1182.

Hecht, R., Kunze, C., and Hahmann, S. (2013). Measuring Completeness of Building Footprints in OpenStreetMap over Space and Time. *ISPRS International Journal of Geo-Information*, 2(4):1066–1091.

Hecht, R., Meinel, G., and Buchroithner, M. (2015). Automatic identification of building types based on topographic databases – a comparison of different data sources. *International Journal of Cartography*, 1(1):18–31.

Henn, A., Römer, C., Gröger, G., and Plümer, L. (2012). Automatic classification of building types in 3D city models. *GeoInformatica*, 16(2):281–306.

Hill, R. C. and Adkins, L. C. (2003). *Collinearity*, pages 256–278. Blackwell Publishing Ltd.

Huber, P. J. (1981). *Robust Statistics*. John Wiley & Sons.

Hutcheson, G. D. and Sofroniou, N. (1999). *The Multivariate Social Scientist : Introductory Statistics Using Generalized Linear Models*. SAGE Publications, 1st edition.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R*, volume 103. Springer, New York, NY.

JAXA (2020). *ALOS Global Digital Surface Model (DSM), ALOS World 3D-30m (AW3D30) Version 3.1, Product Description*. Japan Aerospace Exploration Agency Earth Observation Research Center (JAXA EORC). `https://www.eorc.jaxa.jp/ALOS/en/aw3d30/aw3d30v31_product_e.pdf`.

Jović, A., Brkić, K., and Bogunović, N. (2015). A review of feature selection methods with applications. In *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1200–1205.

Kaden, R. and Kolbe, T. H. (2014). Simulation-Based Total Energy Demand Estimation of Buildings using Semantic 3D City Models. *International Journal of 3-D Information Modeling (IJ3DIM)*, 3(2):35–53.

Kamptner, E. (2020). Metadata Building Footprints. `https://github.com/CityOfNewYork/nyc-geo-metadata/blob/master/Metadata/Metadata_BuildingFootprints.md` (accessed: 10.05.2020).

Kennedy, M. and Kopp, S. (2000). *Understanding Map Projections: GIS by ESRI*. ESRI.

Kontokosta, C. E. (2013). Tall Buildings and Urban Expansion: Tracing the Evolution of Zoning in the United States. *Leadership and Management in Engineering*, 13(3):190–198.

Langley, R. B. (1998). The UTM Grid System. *GPS World*, 9(2):46–50.

Ledoux, H., Ohori, K. A., Kumar, K., Dukai, B., Labetski, A., and Vitalis, S. (2019). CityJSON: a compact and easy-to-use encoding of the CityGML data model. *Open Geospatial Data, Software and Standards*, 4(1-12).

Li, M., Koks, E., Taubenböck, H., and [van Vliet], J. (2020). Continental-scale mapping and analysis of 3D building structure. *Remote Sensing of Environment*, 245:111859.

Liasis, G. and Stavrou, S. (2016). Satellite images analysis for shadow detection and building height estimation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 119:437–450.

Liaw, A. and Wiener, M. (2002). Classification and Regression by randomForest. *R News*, 2(3):18–22.

Lomax, R. G. and Hans-Vaughn, D. L. (2013). *Statistical Concepts: A Second Course*. Routledge, 4th edition.

Lonergan, C. and Hedley, N. (2016). Unpacking isovists: a framework for 3D spatial visibility analysis. *Cartography and Geographic Information Science*, 43(2):87–102.

Meijer, J. R., Huijbregts, M. A. J., Schotten, K. C. G. J., and Schipper, A. M. (2018). Global patterns of current and future road infrastructure. *Environmental Research Letters*, 13(6):064006.

Microsoft (2018). Computer generated building footprints for the United States. `https://github.com/Microsoft/USBuildingFootprints` (accessed: 12.05.2020).

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, New York, NY.

Murphy, R. E. (1972). *The Central Business District*. Routledge, New York.

Nembrini, S., König, I. R., and Wright, M. N. (2018). The revival of the Gini importance? *Bioinformatics*, 34(21):3711–3718.

Nüst, D., Granell, C., Hofer, B., Konkol, M., Ostermann, F. O., Sileryte, R., and Cerutti, V. (2018). Reproducible research and GIScience: an evaluation using AGILE conference papers. *PeerJ*, 6:e5072.

NYC OpenData (2020). Building Footprints. `https://data.cityofnewyork.us/Housing-Development/Building-Footprints/nqwf-w8eh` (accessed: 10.05.2020.

Opitz, D. and Maclin, R. (1999). Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research (JAIR)*, 11.

Osborne, J. W. and Waters, E. (2002). Four assumptions of multiple regression that researchers should always test. *Practical Assessment, Research, and Evaluation*, 8.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

QGIS Development Team (2020). QGIS - A Free and Open Source Geographic Information System. `https://qgis.org/en/site/` (accessed: 12.05.2020).

Rappaport, J. (2008). Consumption amenities and city population density. *Regional Science and Urban Economics*, 38(6):533–552.

Rinckes, D. and Bung, P. (2019). Open Location Code: An Open Source Standard for Addresses, Independent of Building Numbers And Street Names. `https://github.com/google/open-location-code/blob/master/docs/olc_definition.adoc` (accessed: 16.04.2020).

Ross, L. (2010). *Virtual 3D City Models in Urban Land Management - Technologies and Applications*. PhD thesis, Technische Universität Berlin, Berlin, Germany.

Shao, Y., Taff, G. N., and Walsh, S. J. (2011). Shadow detection and building-height estimation using IKONOS data. *International Journal of Remote Sensing*, 32(22):6929–6944.

Smith, B. and Sandwell, D. (2003). Accuracy and resolution of shuttle radar topography mission data. *Geophysical Research Letters*, 30(9).

Smola, A. J. and Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, 14:199–222.

Stoter, J., de Kluijver, H., and Kurakula, V. (2008). 3D noise mapping in urban areas. *International Journal of Geographical Information Science*, 22(8):907–924.

Strobl, C. (2008). *PostGIS*, pages 891–898. Springer US, Boston, MA.

Sun, Z., Fang, H., Deng, M., Chen, A., Yue, P., and Di, L. (2015). Regular Shape Similarity Index: A Novel Index for Accurate Extraction of Regular Objects From Remote Sensing Images. *IEEE Transactions on Geoscience and Remote Sensing*, 53:3737–3748.

Sutton, R. and Barto, A. (1998). *Reinforcement Learning: An Introduction*. MIT Press.

The PostgreSQL Global Development Group (1996–2020). Chapter 29. Reliability and the Write-Ahead Log. PostgreSQL 11.7 Documentation: `https://www.postgresql.org/docs/11/wal-intro.html` (accessed: 12.05.2020).

TheKernelTrip (2018). Computational complexity of machine learning algorithms. `https://www.thekerneltrip.com/machine/learning/computational-complexity-learning-algorithms/` (accessed: 03.01.2020).

Toloşi, L. and Lengauer, T. (2011). Classification with correlated features: unreliability of feature ranking and solutions. *Bioinformatics*, 27(14):1986–1994.

U.S. Geological Survey (2017a). 1 Arc-second Digital Elevation Models (DEMs) - USGS National Map 3DEP Downloadable Data Collection. `https://www.sciencebase.gov/catalog/item/4f70aa71e4b058caae3f8de1` (accessed: 10.05.2020).

U.S. Geological Survey (2017b). What is the State Plane Coordinate System? Can GPS provide coordinates in these values? `https://www.usgs.gov/faqs/what-state-plane-coordinate-system-can-gps-provide-coordinates-these-values?qt-news_science_products=0#qt-news_science_products` (accessed: 15.12.2019).

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Number 2. Springer, New York.

Wang, N., Vlachokostas, A., Borkum, M., Bergmann, H., and Zaleski, S. (2019). Unique Building Identifier: A natural key for building data matching and its energy applications. *Energy and Buildings*, 184:230–241.

Warmerdam, F., Rouault, E., et al. (2019). ogr2ogr - GDAL Documentation. `https://gdal.org/programs/ogr2ogr.html` (accessed: 03.12.2019).

Weisberg, S. (2005). *Applied Linear Regression*. John Wiley & Sons, Inc., 3rd edition.

Zhu, Z., Ong, Y.-S., and Dash, M. (2007). Wrapper-Filter Feature Selection Algorithm Using a Memetic Framework. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(1):70–76.

Zillow (2018). Zillow - US Neighborhoods. `https://data.opendatasoft.com/explore/dataset/zillow-neighborhoods%40public/information/` (accessed: 10.05.2020).

Colophon

This document was typeset using LaTeX, using the KOMA-Script class `scrbook`. The main font is Palatino.